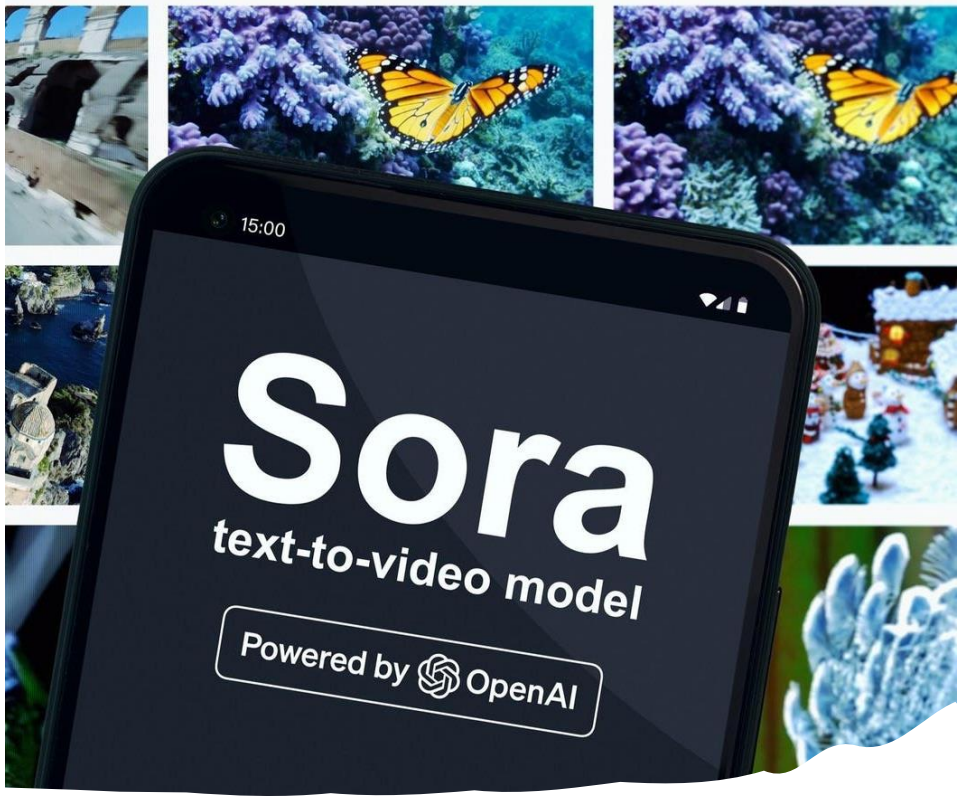
Diffusion Distillation in 2025

Tianwei Yin
Reve AI

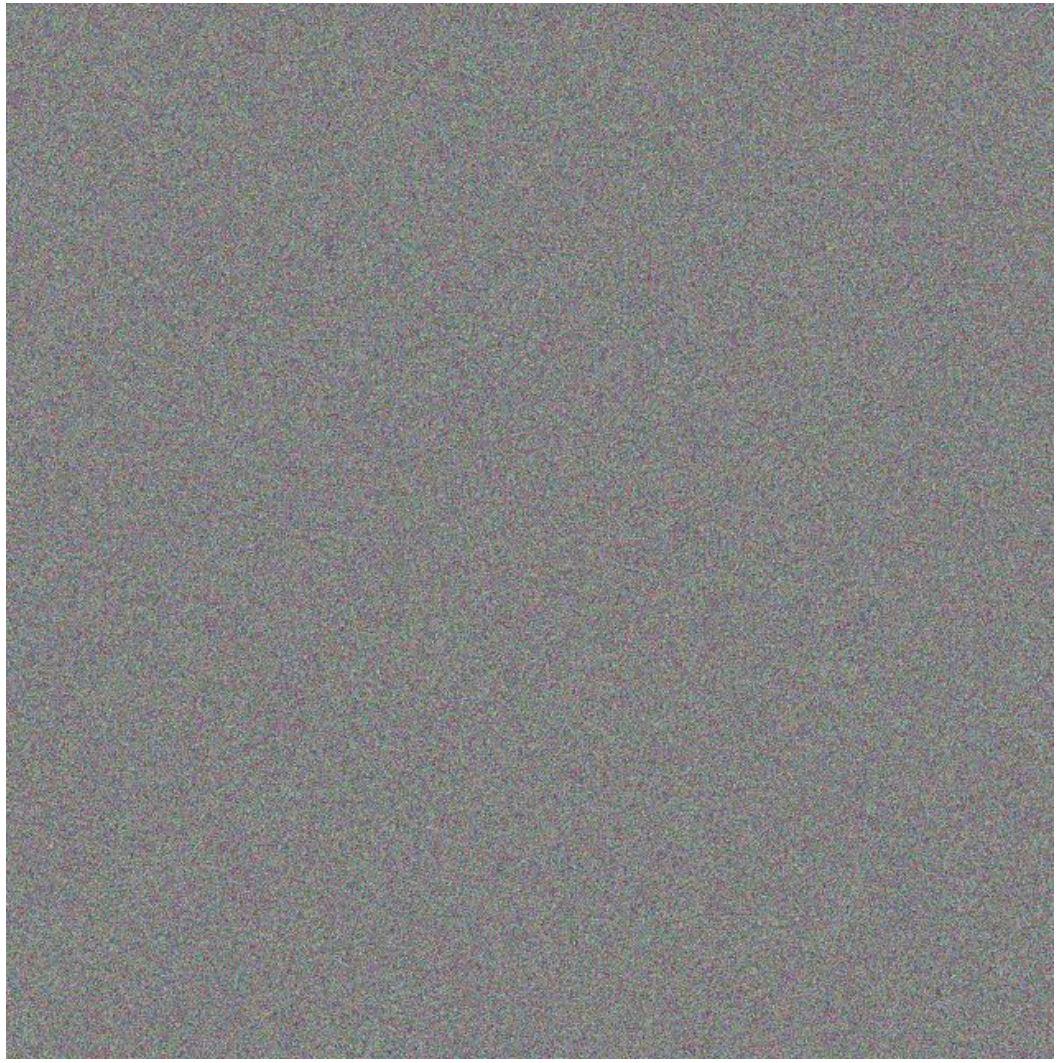
OpenAI launches Dall-E 3

text-to-image



Recent Advances in Diffusion-based Generative Models

Sampling with Diffusion Models is Slow



- Orders of magnitude slower than GANs and VAEs
- Iterative denoising process
  - 20-50 steps for reasonable images
- We speed it up by step distillation!

10s / Image
50-step Teacher

0.8s / Image
4-step Student

Overview of Diffusion Distillation

Trajectory Preserving

Knowledge Distillation [LL, 2020]

Progressive Distillation [SH, ICLR 2022]

Guidance Distillation [MRGKEHS, CVPR 2023]

Consistency Models [SDCS, ICML 2023]

Distribution Matching

DMD [YGZSDFP, CVPR 2024]

DMD 2 [YGPZSDF, NeurIPS 2024]

Diff-Instruct [LHZSLZ, NeurIPS 2023]

MMD [SMHH, NeurIPS 2024]

CausVid [YZZFDSH, CVPR 2025]

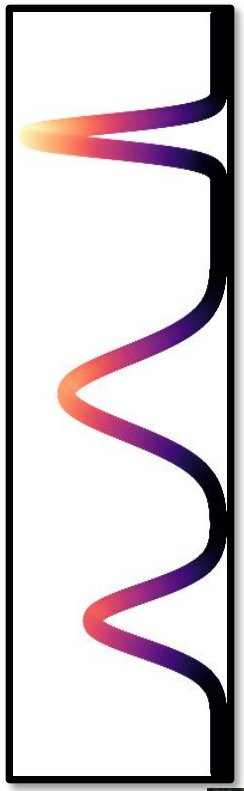
GAN

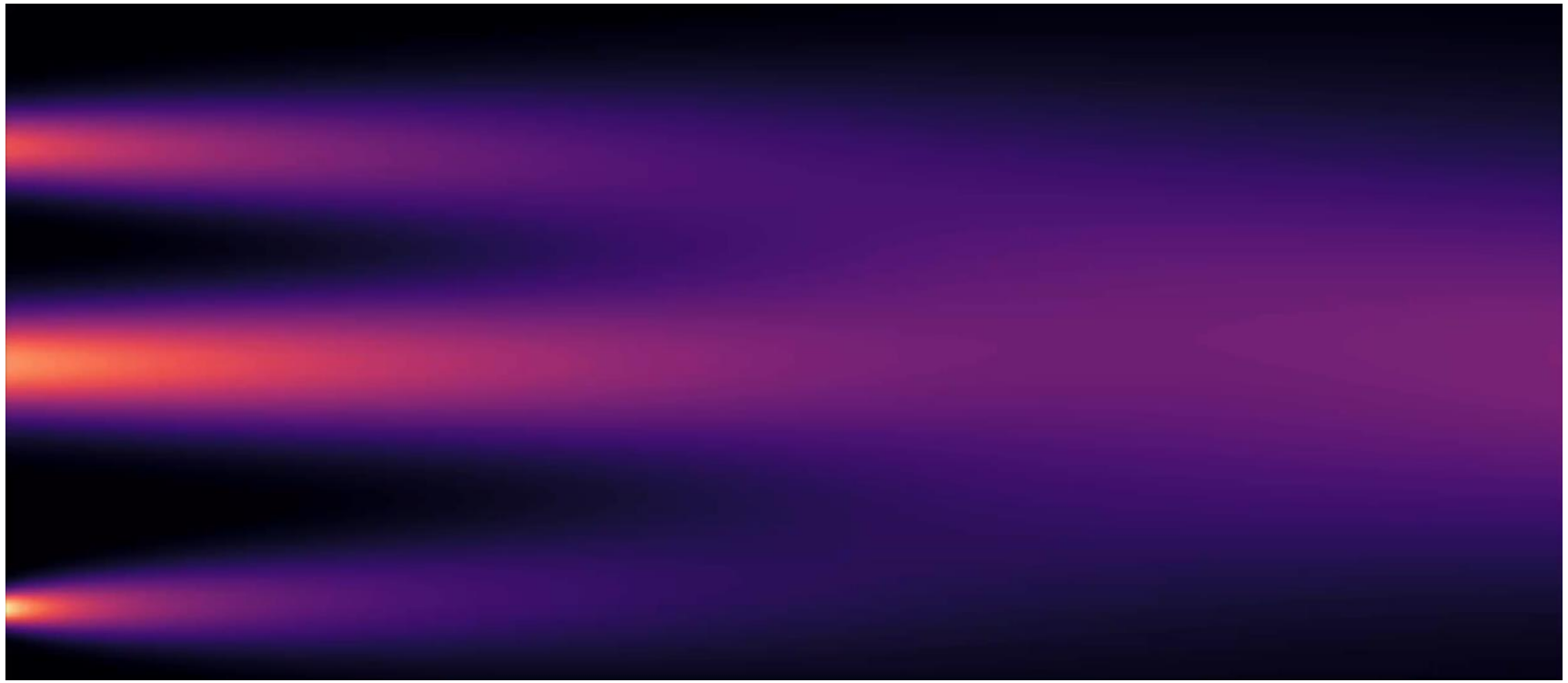
Too many, covered in Junyan's talk.

Diffusion Basics

Converting Data Distribution to Gaussian

$p_t(x)$



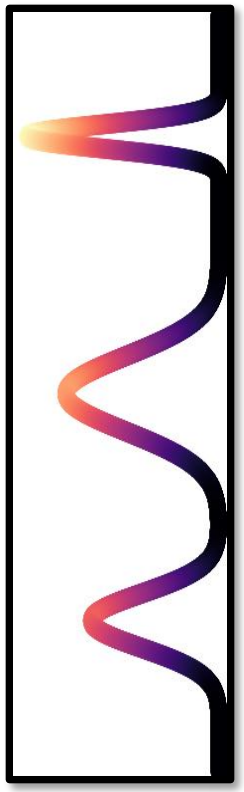


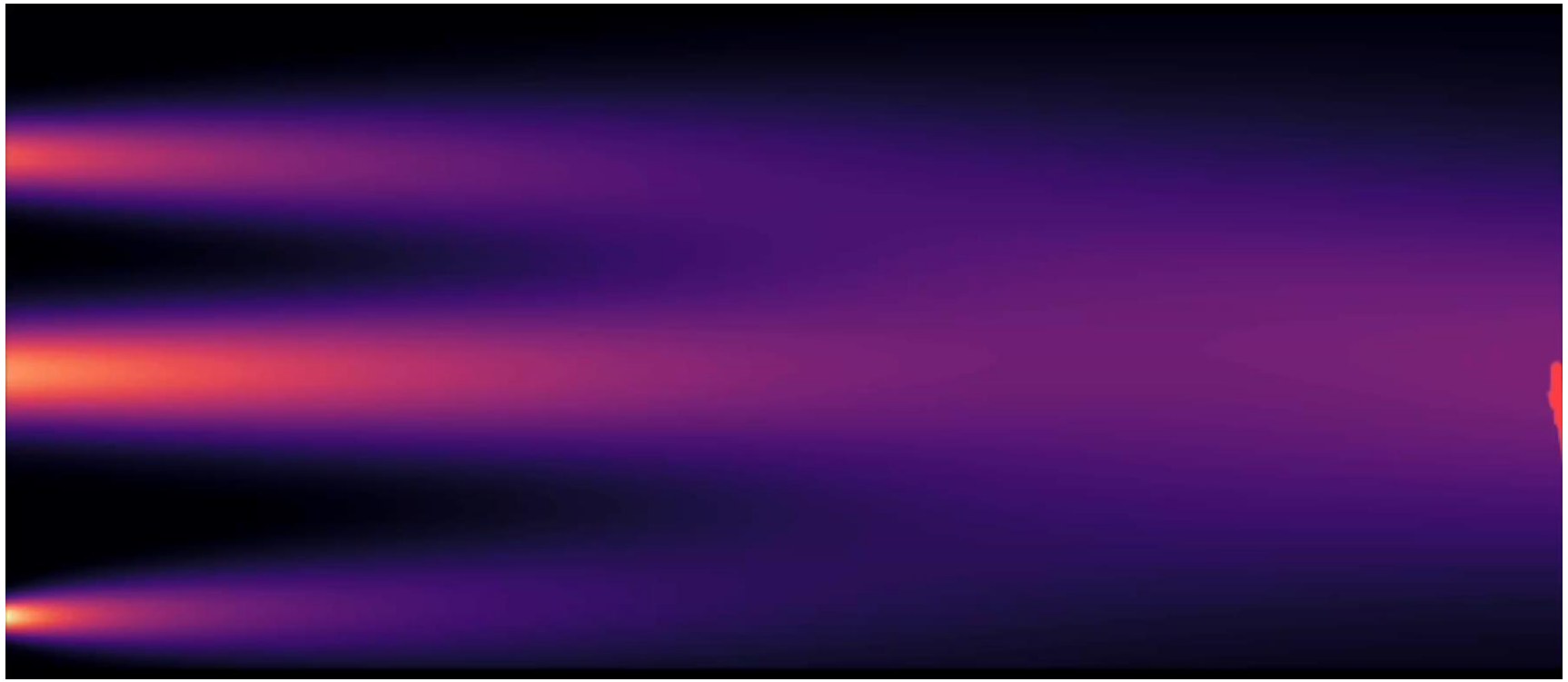
$p_0(x) = p_{data}(x)$

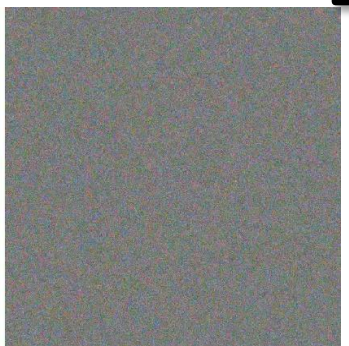
$p_T(x) \approx \mathcal{N}(0,1)$

Generating Sample from Noise

$p_t(x)$



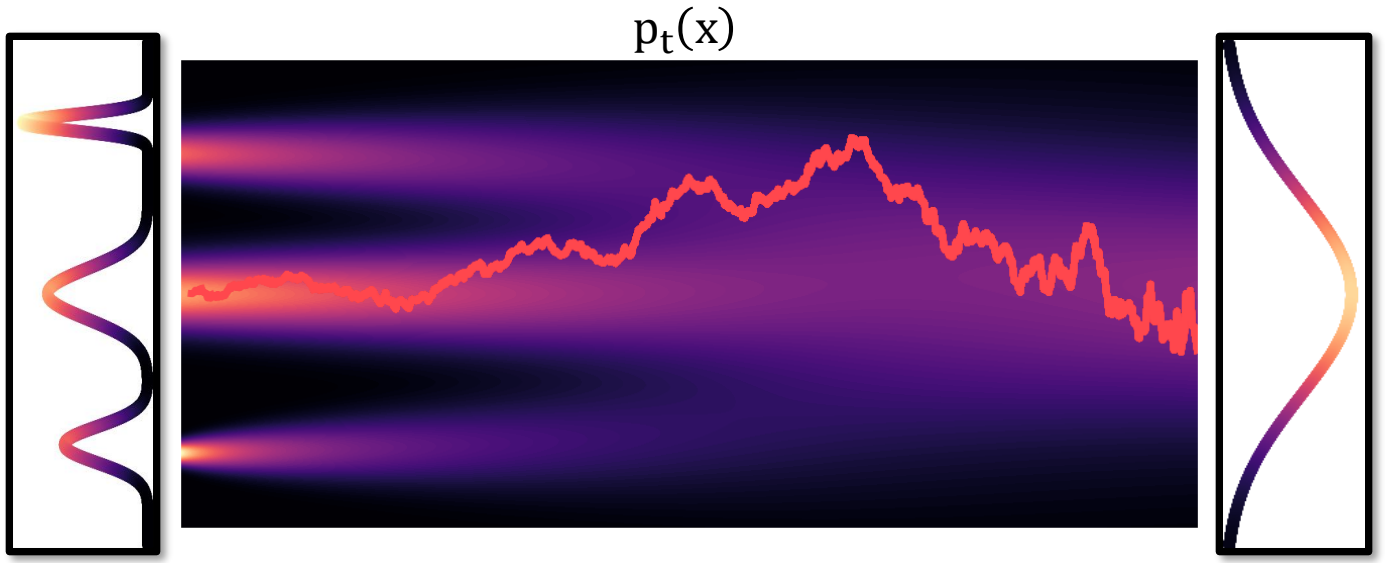




$p_0(x)$
$=$
$p_{data}(x)$

$p_T(x)$
$\approx$
$\mathcal{N}(0,1)$

Basics of Diffusion Model



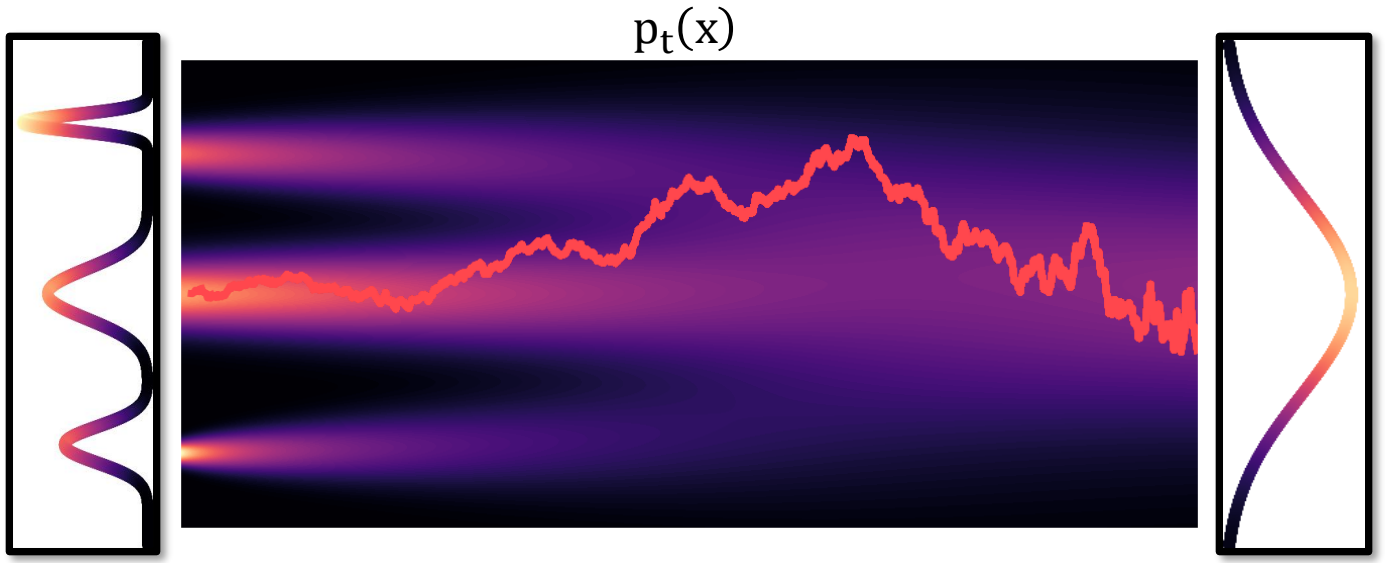
Forward Process (data to noise)

$$dx_t = \sigma_t dw_t$$

Backward Process (noise to data)

$$dx_t = -\sigma_t^2 \nabla_x \log p_t(x) dt + \sigma_t dw_t$$

Basics of Diffusion Model



Backward Process (noise to data)

$$dx_t = -\sigma_t^2 \nabla_x \log p_t(x) dt + \sigma_t dw_t$$

score function

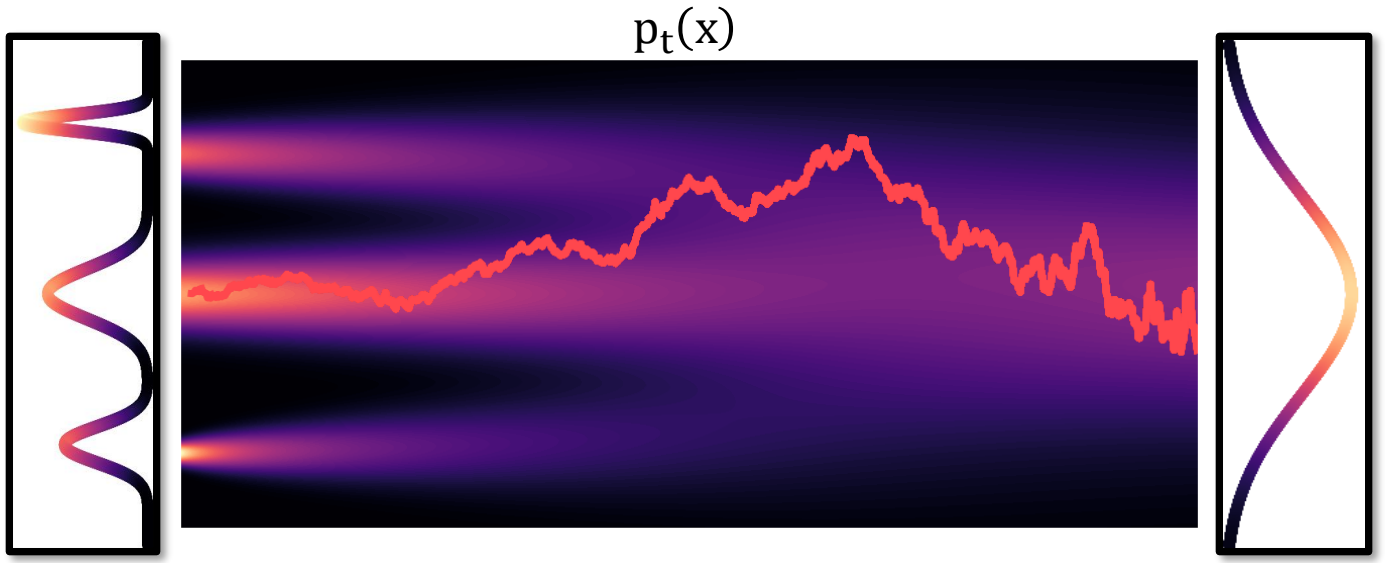
$s_\theta(x,t) = \nabla_x \log p_t(x)$

Neural Network

Score Matching Loss:

$$\mathbb{E}_{t,x}[\lambda(t)\|\nabla_x \log p_t(x) - s_\theta(x,t)\|]$$

Basics of Diffusion Model



Backward Process (noise to data)

$$dx_t = -\sigma_t^2 \; s_\theta(x,t)\,dt + \sigma_t dw_t$$

If we omit the intermediate noise injection ($\sigma_t dw_t$), we obtain a backward ODE that establishes one-to-one mappings between noise and data.

We can utilize any ODE solvers (DDIM, Euler, etc.)

Overview of Diffusion Distillation

Trajectory Preserving

Knowledge Distillation [LL, 2020]

Progressive Distillation [SH, ICLR 2022]

Guidance Distillation [MRGKEHS, CVPR 2023]

Consistency Models [SDCS, ICML 2023]

Distribution Matching

DMD [YGZSDFP, CVPR 2024]

DMD 2 [YGPZSDF, NeurIPS 2024]

Diff-Instruct [LHZSLZ, NeurIPS 2023]

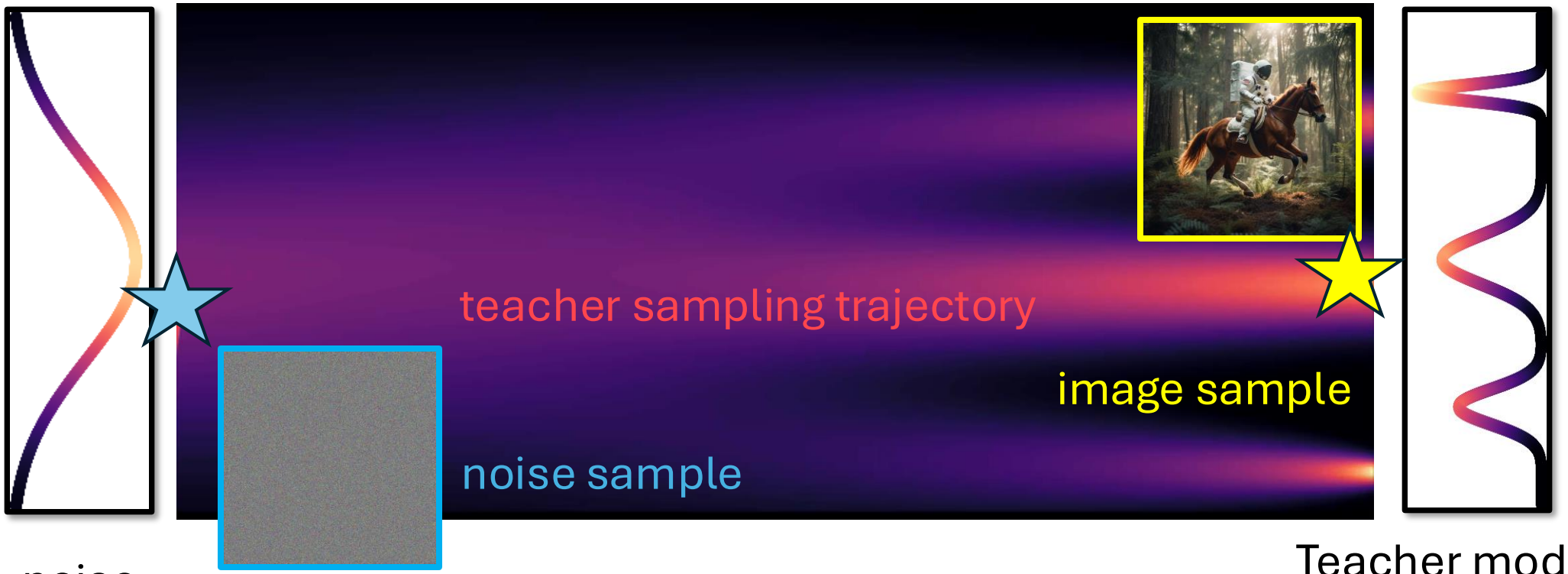
MMD [SMHH, NeurIPS 2024]

CausVid [YZZFDSH, CVPR 2025]

GAN

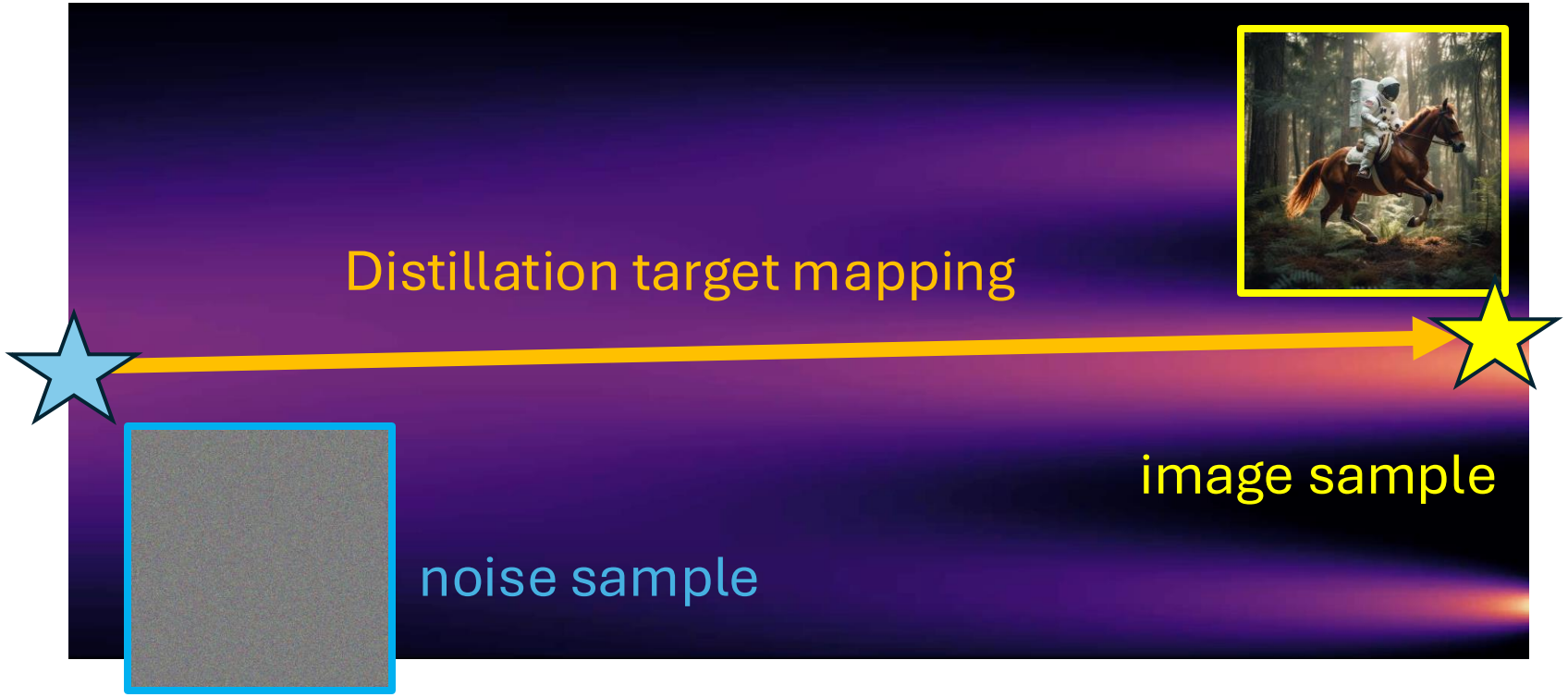
Too many, covered in Junyan's talk.

Diffusion distillation, its challenges



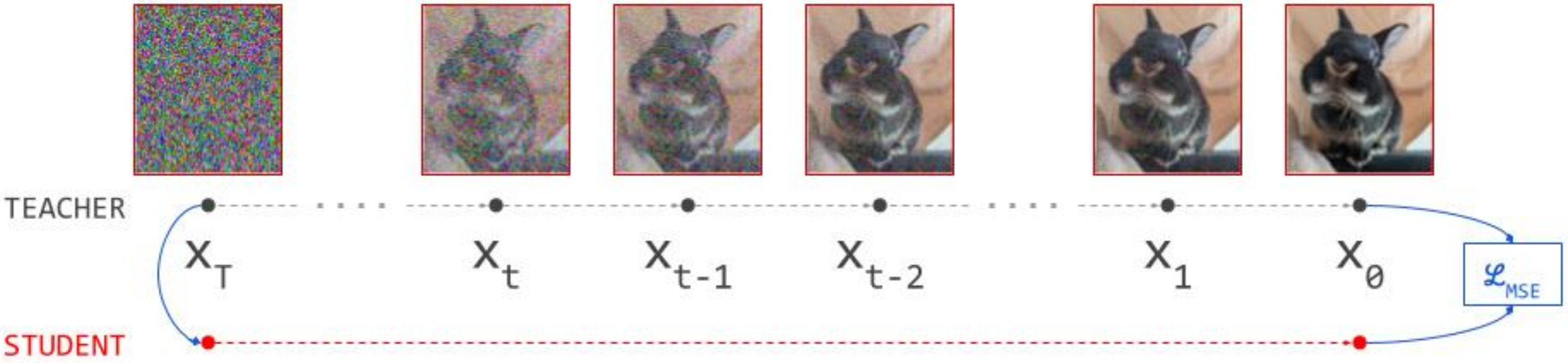
Luhman et al. Knowledge Distillation. ArXiv 2021
Salimans et al. Progressive Distillation for Fast Sampling of Diffusion Models. ICLR 2022
Song et al. Consistency Models. ICML 2023

Trajectory-preserving distillation tries to recover **pointwise mapping** from noise-to-image



Luhman et al. Knowledge Distillation. ArXiv 2021
Salimans et al. Progressive Distillation for Fast Sampling of Diffusion Models. ICLR 2022
Song et al. Consistency Models. ICML 2023

Knowledge Distillation: Distilling diffusion sampling into a single forward pass

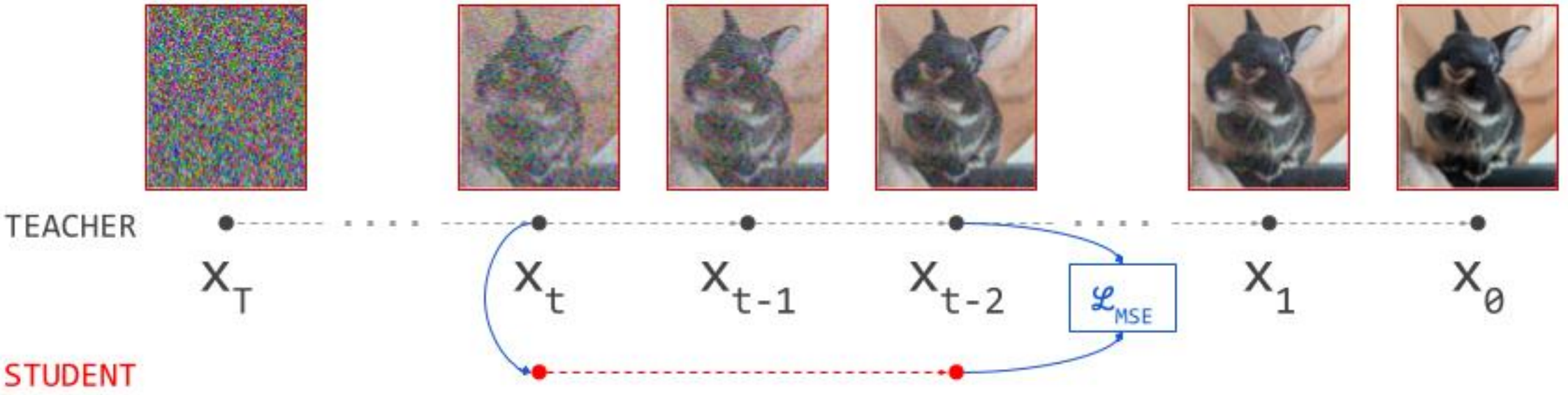


Distill the entire sampling procedure into a network with the same architecture used for a single diffusion prediction step, by matching outputs with a MSE loss

Proof-of-Concept: We really don't need 50 or 1000 steps!
Slow to Run: Requires full trajectory sampling
Low Performance Upper Bound: Error accumulation in deterministic sampling

Luhman et al. Knowledge Distillation. ArXiv 2021
Image credit: https://sander.ai/2024/02/28/paradox.html

Progressive Distillation: Iteratively halve the number of sampling steps



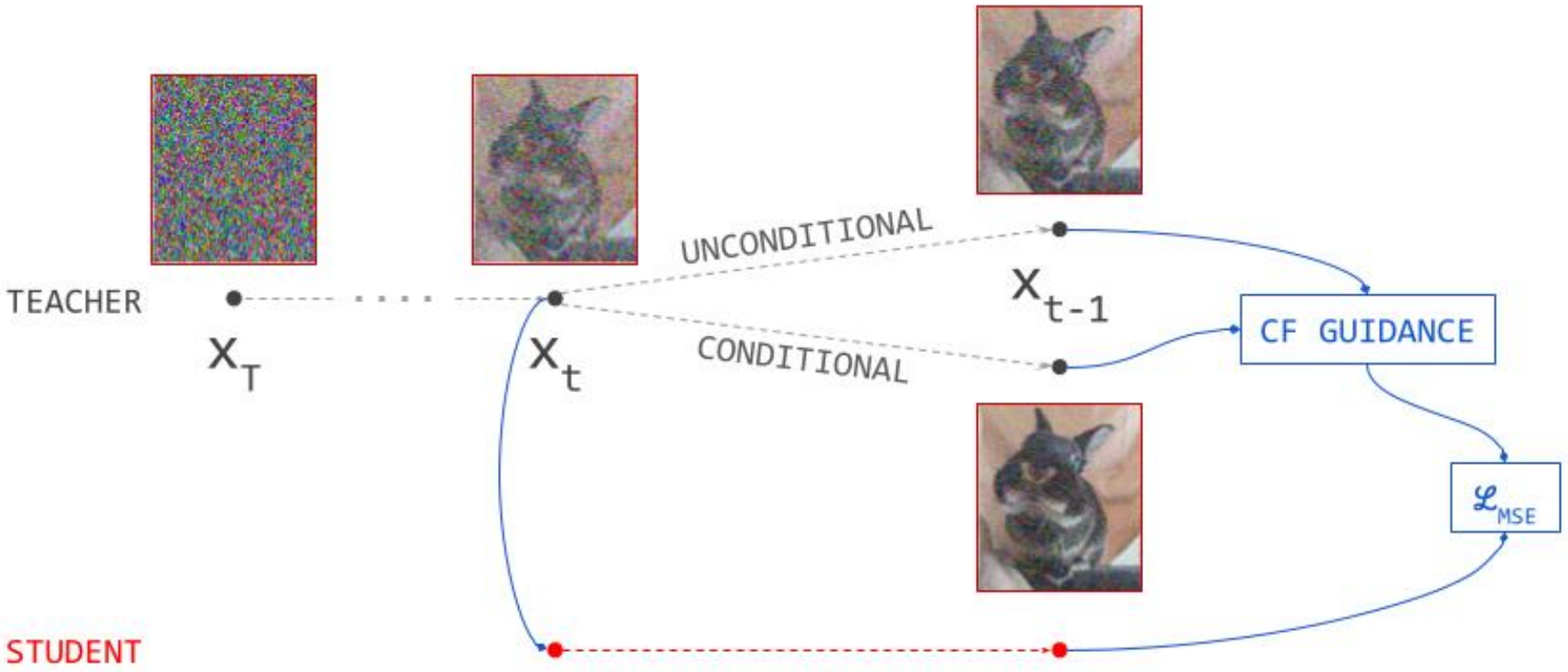
The teacher uses the full diffusion process (e.g., many time steps). The student then learns to produce similar outputs but in half the number of steps.

Faster-to-Run: No precomputation needed!

Better quality tradeoff: 8 step generally leads to satisfactory result

Salimans, Tim, and Jonathan Ho. Progressive distillation for fast sampling of diffusion models.
Image credit: https://sander.ai/2024/02/28/paradox.html

Guidance Distillation: Avoid two network passes in classifier-free guidance



Approximate guided score with a single network forward pass.

Effective and no quality drop.
Used in Flux-dev image generation

Meng, Chenlin, et al. "On distillation of guided diffusion models."CVPR 2023.
Image credit:
https://sander.ai/2024/02/28/paradox.html

Overview of Diffusion Distillation

Trajectory Preserving

Knowledge Distillation [LL, 2020]

Progressive Distillation [SH, ICLR 2022]

Guidance Distillation [MRGKEHS, CVPR 2023]

Consistency Models [SDCS, ICML 2023]

Distribution Matching

DMD [YGZSDFP, CVPR 2024]

DMD 2 [YGPZSDF, NeurIPS 2024]

Diff-Instruct [LHZSLZ, NeurIPS 2023]

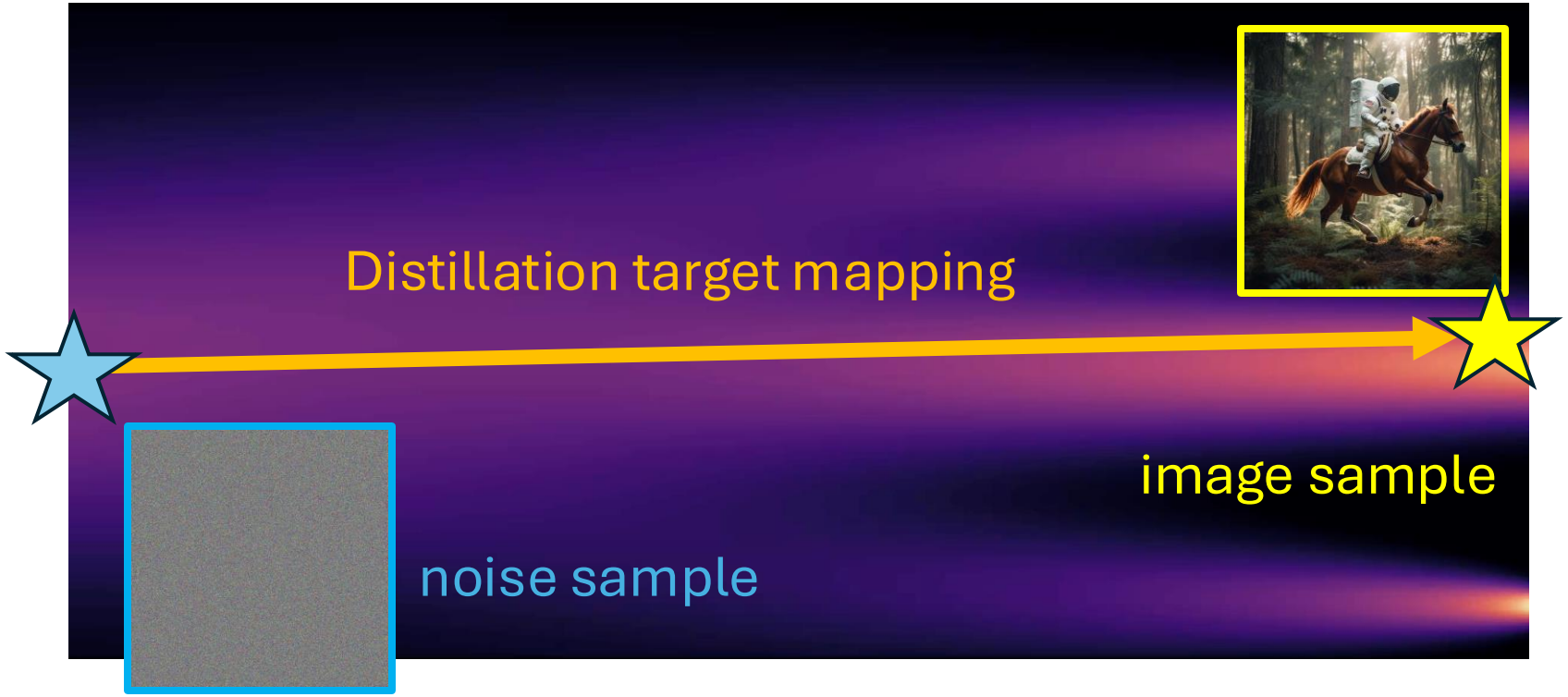
MMD [SMHH, NeurIPS 2024]

CausVid [YZZFDSH, CVPR 2025]

GAN

Too many, covered in Junyan's talk.

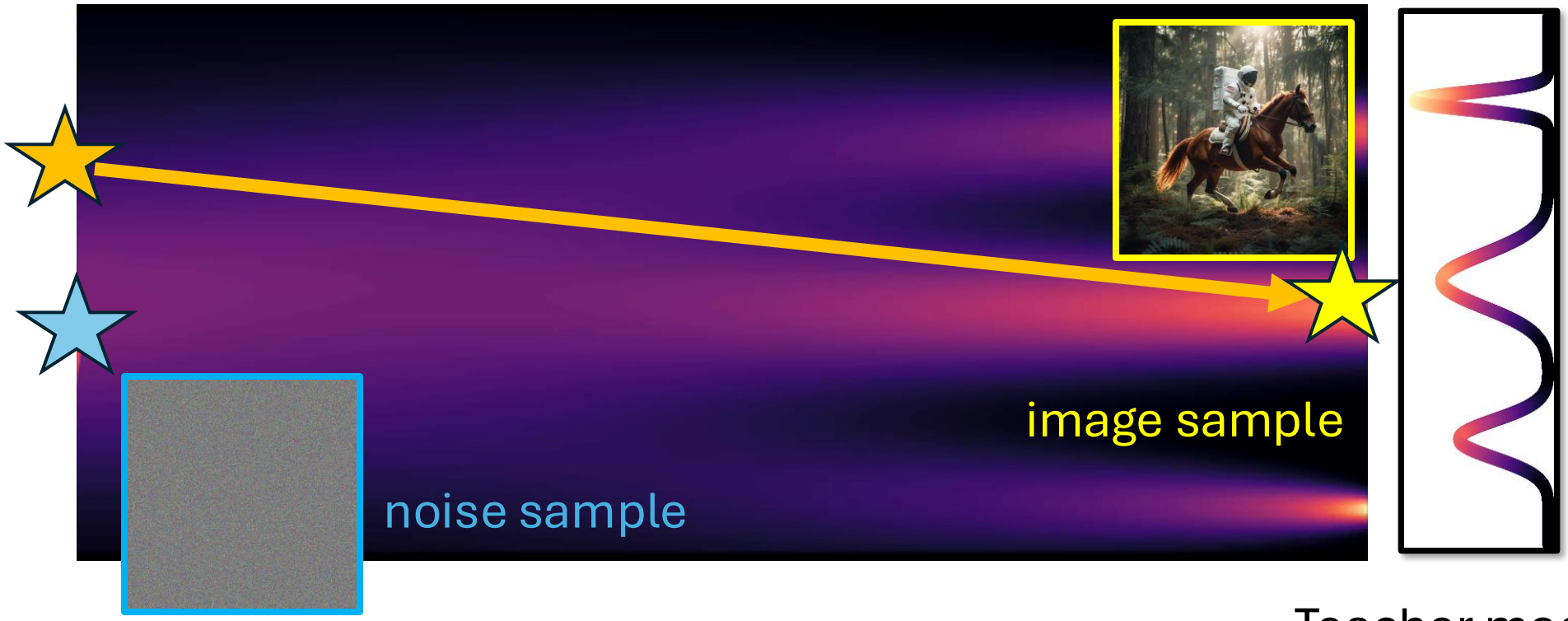
Trajectory-preserving distillation tries to recover **pointwise mapping** from noise-to-image



Luhman et al. Knowledge Distillation. ArXiv 2021
Salimans et al. Progressive Distillation for Fast Sampling of Diffusion Models. ICLR 2022
Song et al. Consistency Models. ICML 2023

DMD matches the teacher's **distribution**

learning the exact noise -> image mapping is unnecessarily hard



Different Mapping, Same Quality, Easier to train

Teacher model's output distribution

KL Divergence Minimization using Diffusion Models

Teacher-generated images

$$D_{KL}\left(p_{\text{fake}} \parallel p_{\text{real}}\right)$$

Student-generated images

$$\nabla_\theta D_{KL} = \mathop{\mathbb{E}}_{\substack{z\sim\mathcal{N}(0;\mathbf{I}) \\ x=G_\theta(z)}} \Big[-\big(s_{\text{real}}(x) - s_{\text{fake}}(x)\big)\, \nabla_\theta G_\theta(z) \Big]$$

$$\text{where } s_{\text{real}}(x) = \nabla_x \log p_{\text{real}}(x),\ s_{\text{fake}}(x) = \nabla_x \log p_{\text{fake}}(x)$$

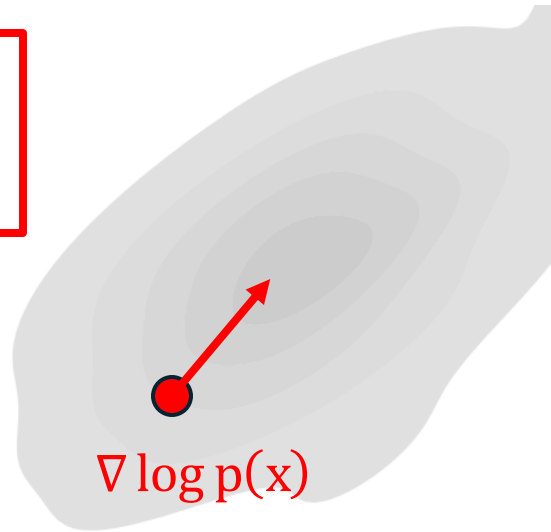
Song et al. ScoreSDE. ICLR 2021

Key insight

We can approximate these two gradients using 2 diffusion models!

$$\nabla_\theta D_{KL} = \mathop{\mathbb{E}}_{\substack{z\sim\mathcal{N}(0;\mathbf{I}) \\ x=G_\theta(z)}} \Big[-\big(s_{\text{real}}(x) - s_{\text{fake}}(x)\big) \nabla_\theta G_\theta(z)$$

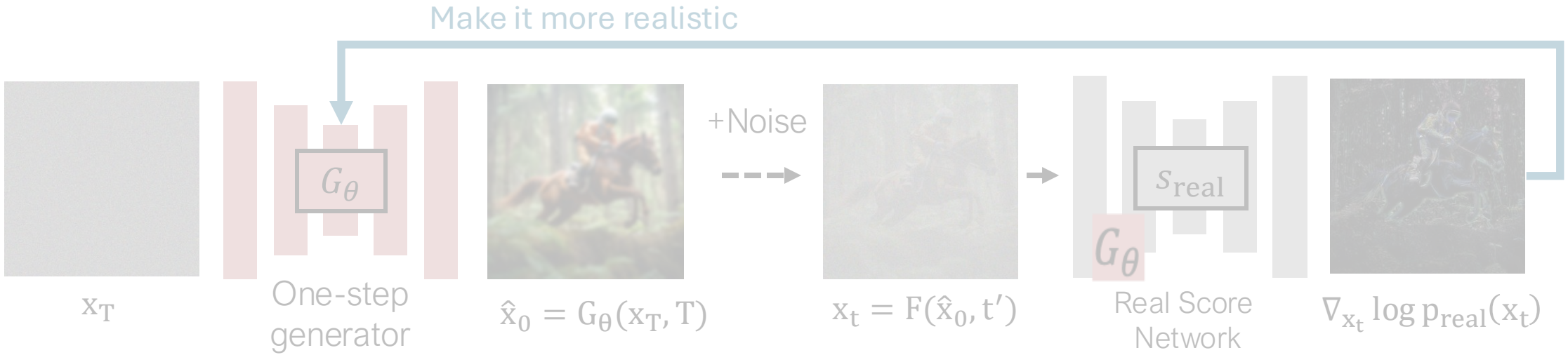
$$\text{where } s_{\text{real}}(x) = \nabla_x \log p_{\text{real}}(x),\ s_{\text{fake}}(x) = \nabla_x \log p_{\text{fake}}(x)$$

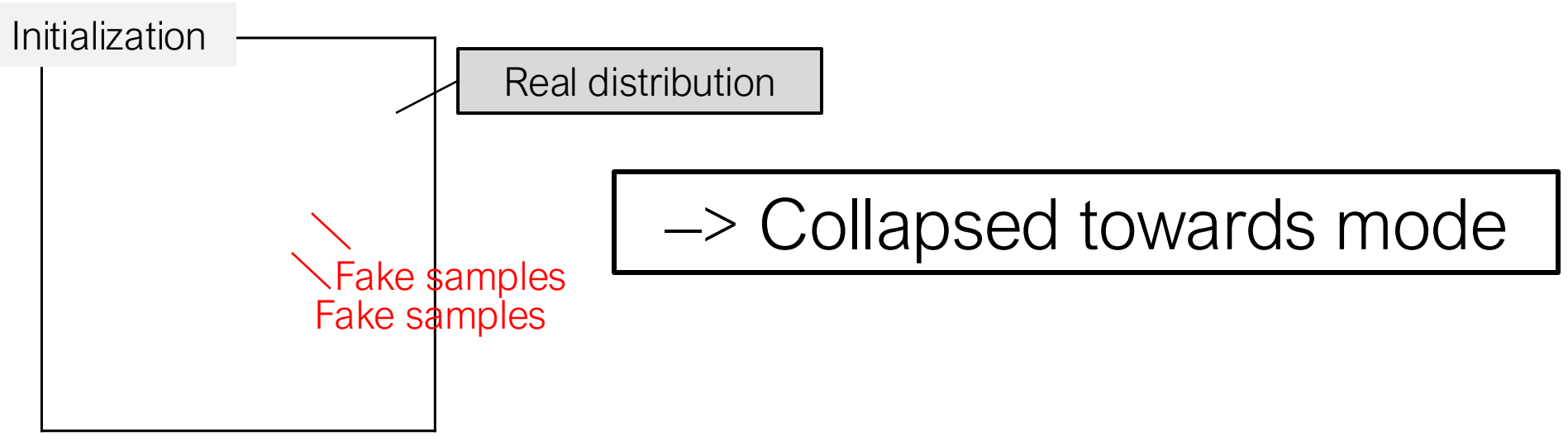


Diffusion Model
~=
Score Function

Song et al. ScoreSDE. ICLR 2021

Can we supervise a 1-step Generator using a single pretrained Score Function?

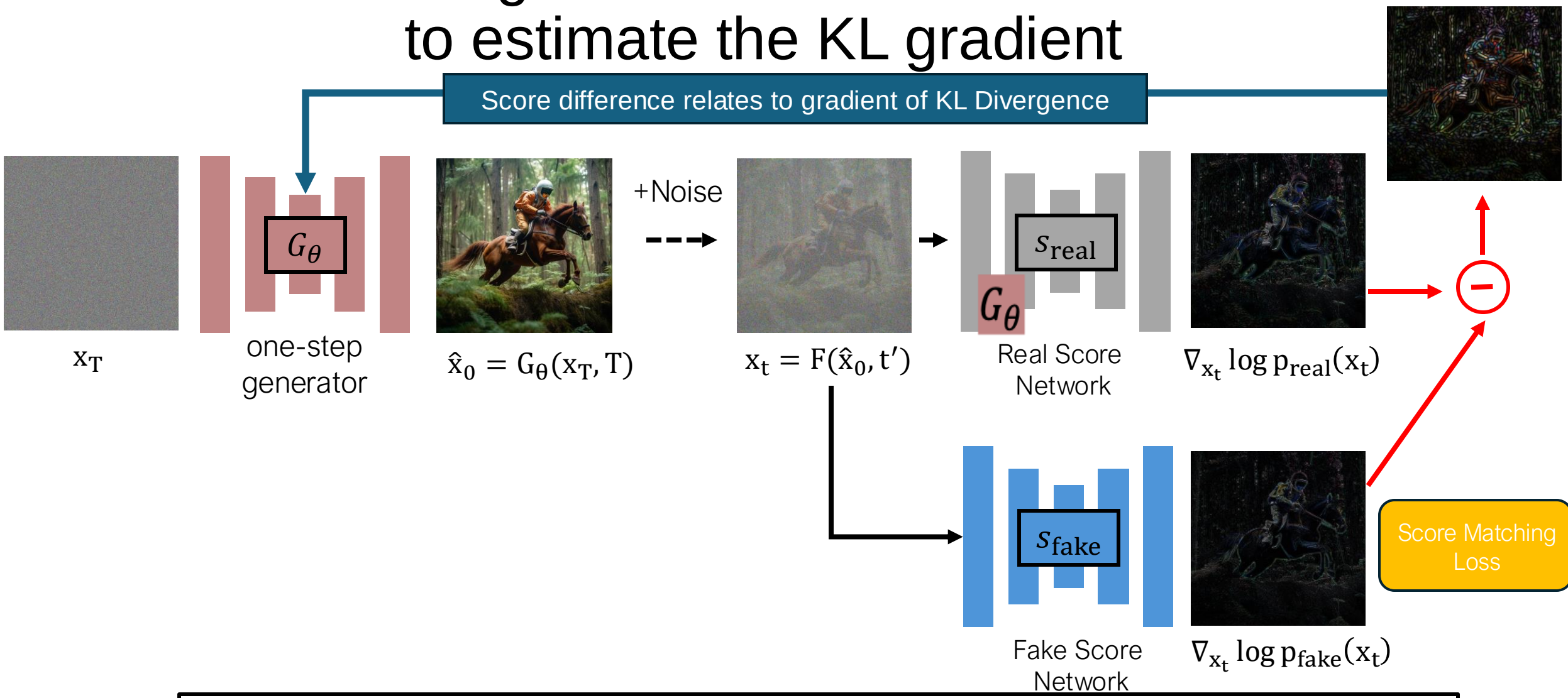




–> Collapsed towards mode

Yin et al. One-step diffusion with distribution matching distillation. CVPR 2024
Wang et al. ProlificDreamer. NeurIPS 2023
Poole et al. DreamFusion. ICLR 2023

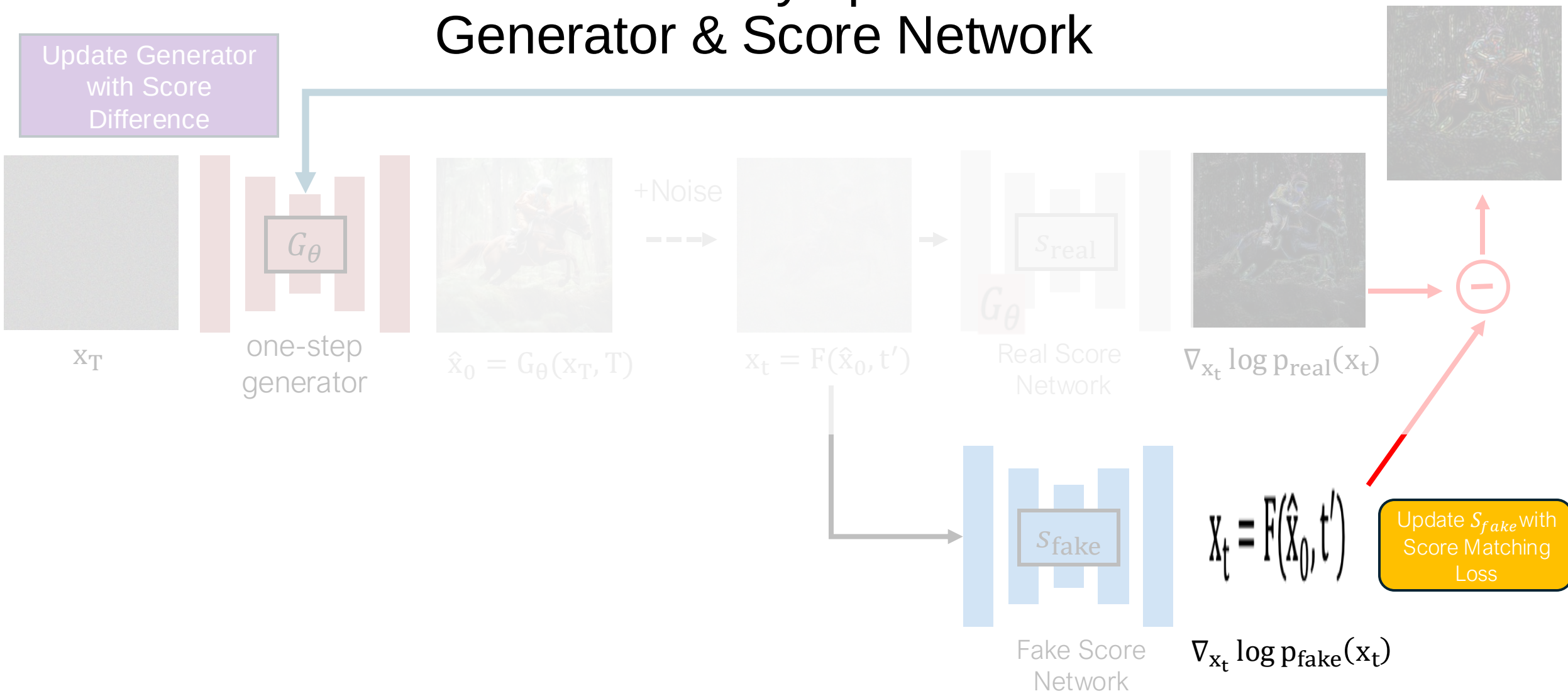
Training a second Score Function to estimate the KL gradient



Train a score function on the *fake/generated* images

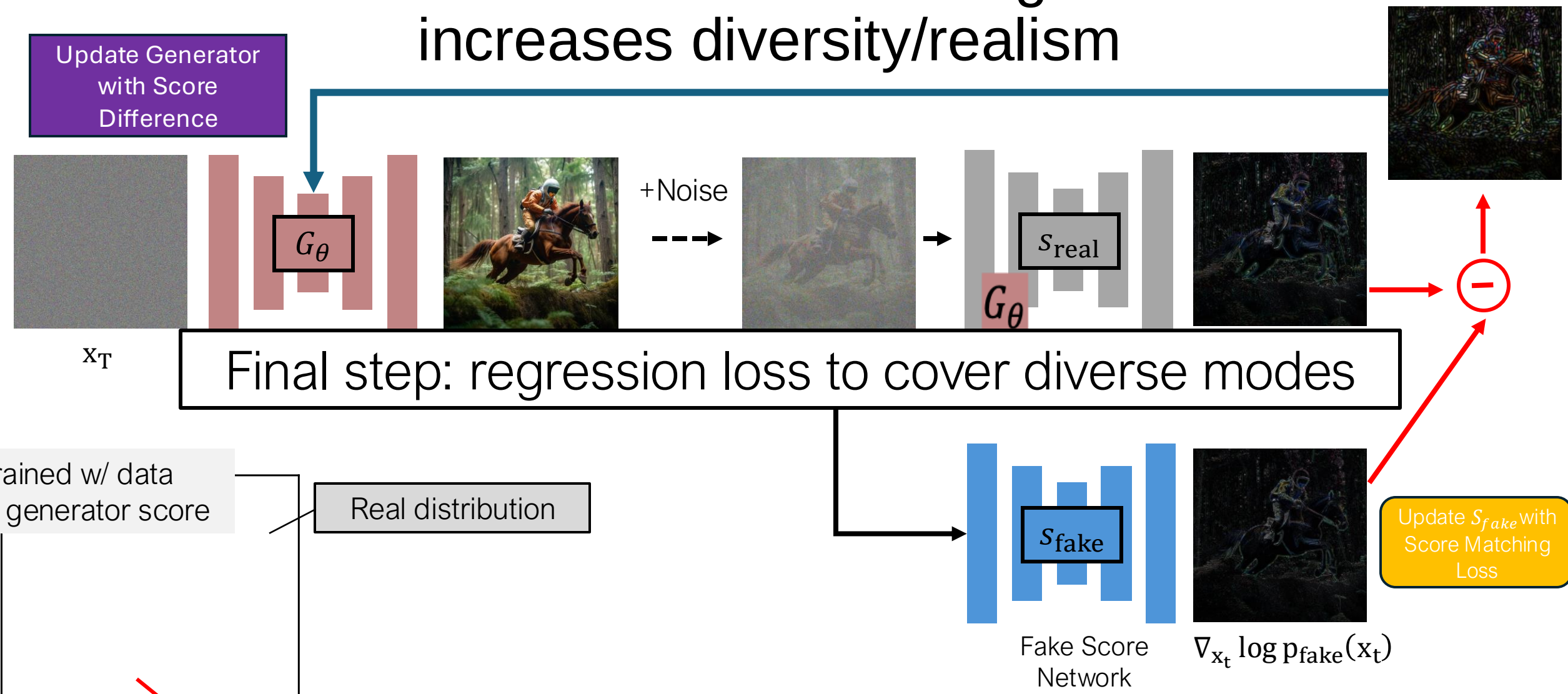
Yin et al. One-step diffusion with distribution matching distillation. CVPR 2024

Alternatively optimize Generator & Score Network



Yin et al. One-step diffusion with distribution matching distillation. CVPR 2024

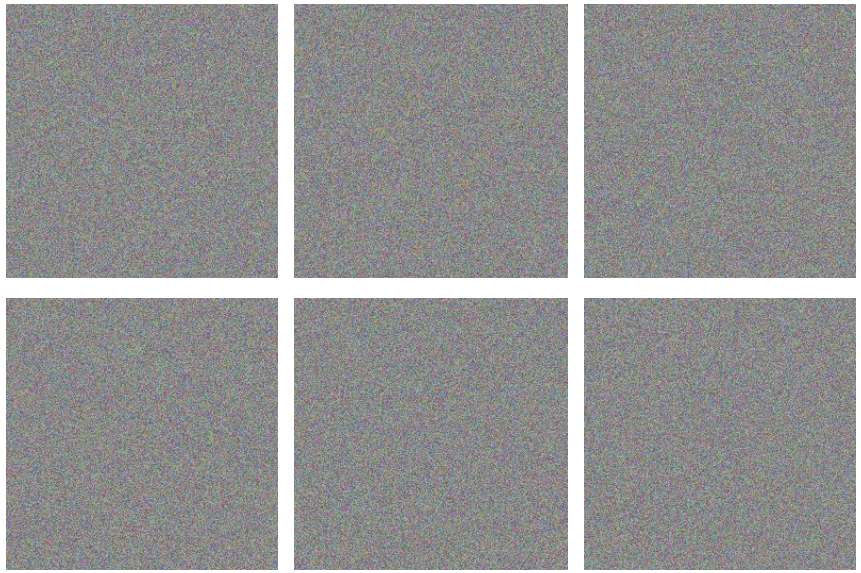
Distribution matching increases diversity/realism



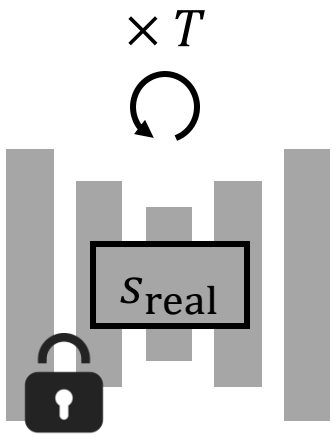
Yin et al. One-step diffusion with distribution matching distillation. CVPR 2024

Luhman et al. Knowledge Distillation. ArXiv 2021

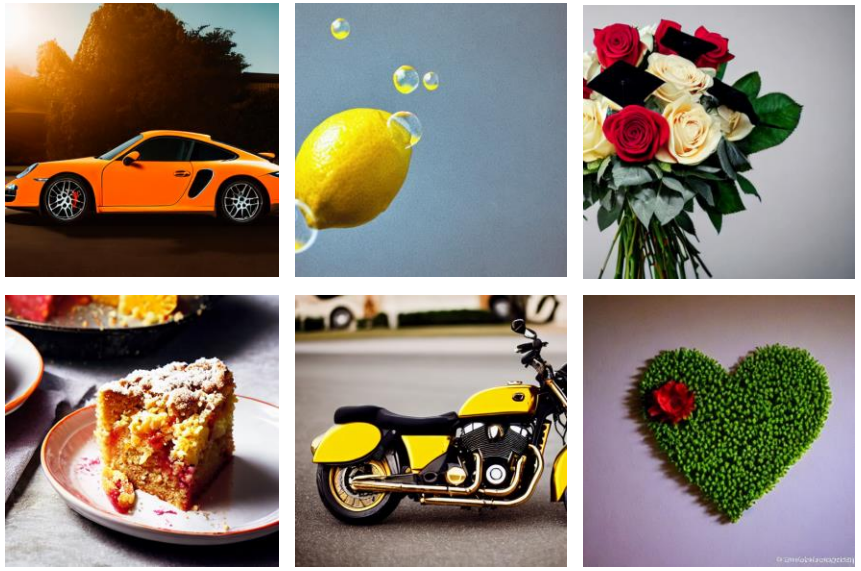
Noise-Image Pair Collection



input noise maps

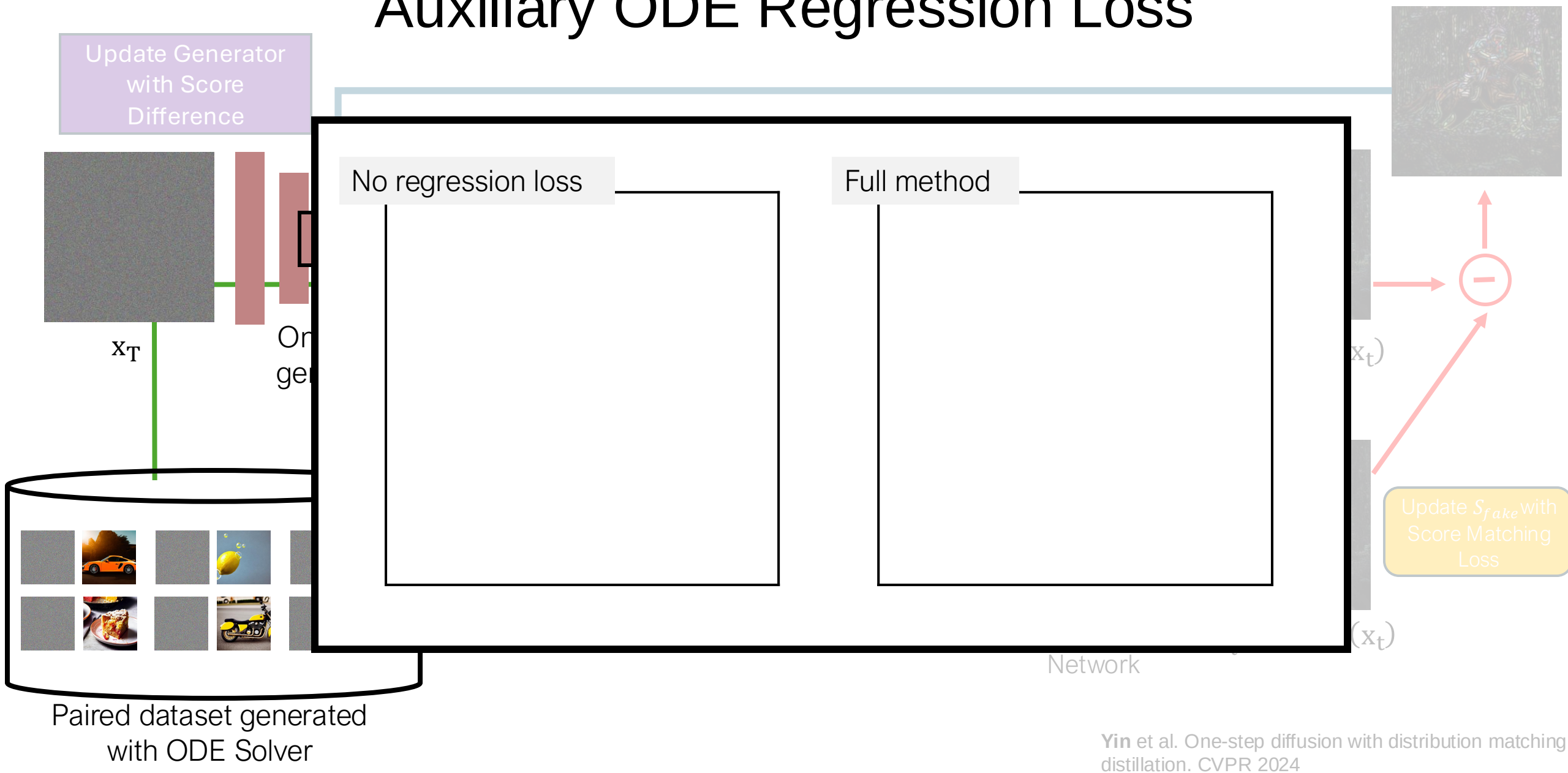


Sample
pretrained mode
with iterative denoising



output images

Auxiliary ODE Regression Loss

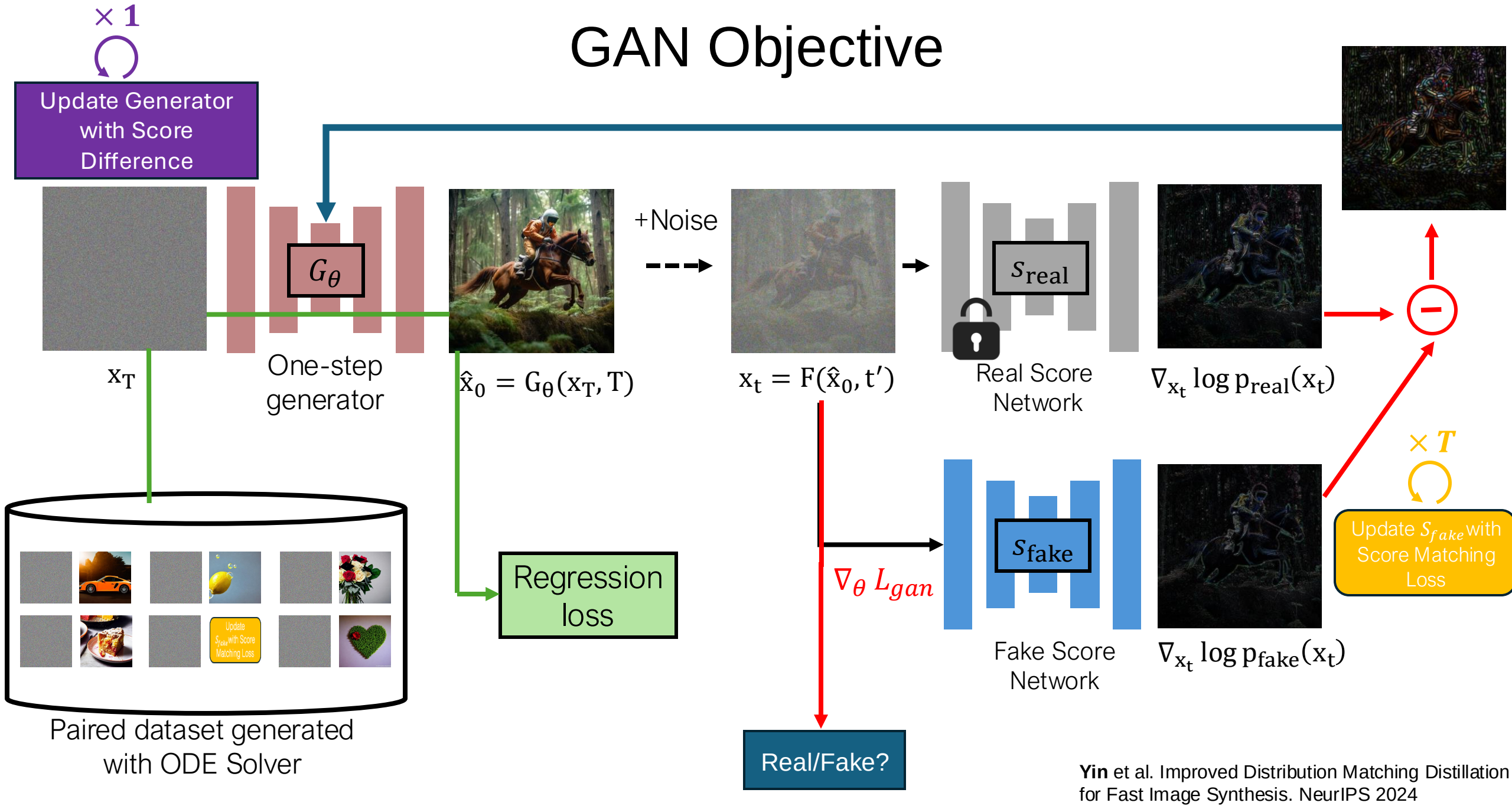


Yin et al. One-step diffusion with distribution matching distillation. CVPR 2024

Luhman et al. Knowledge Distillation. ArXiv 2021

Improvement 1
GAN

GAN Objective



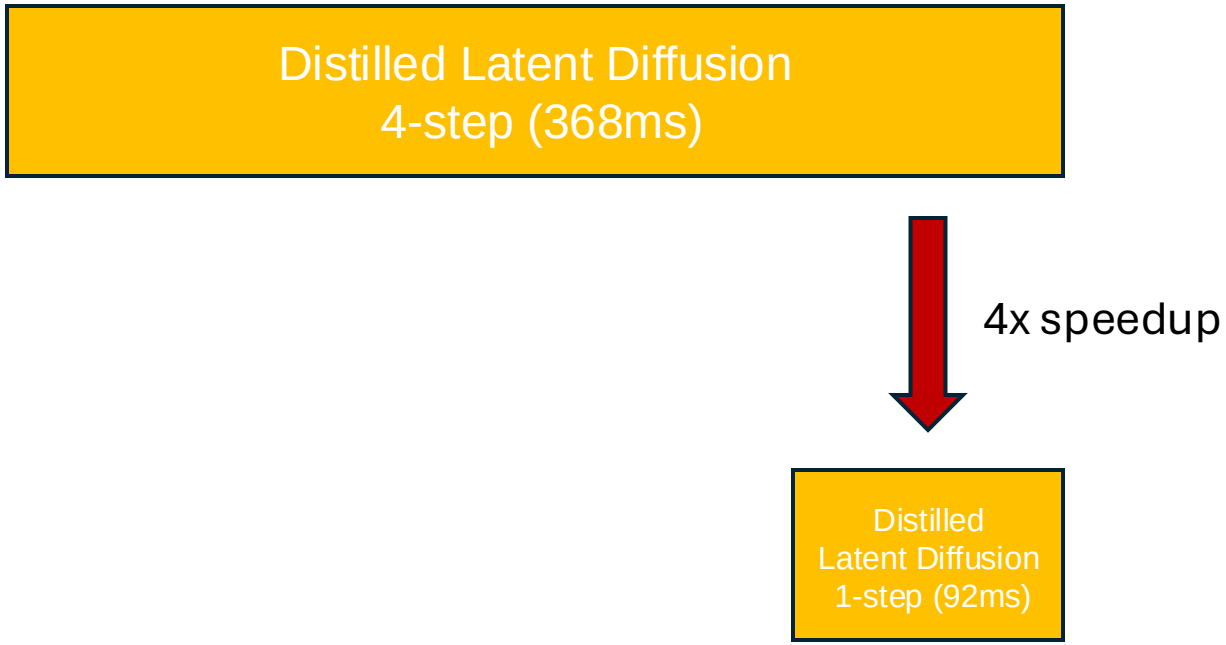
Yin et al. Improved Distribution Matching Distillation for Fast Image Synthesis. NeurIPS 2024


No GAN Loss
Ours w. GAN Loss

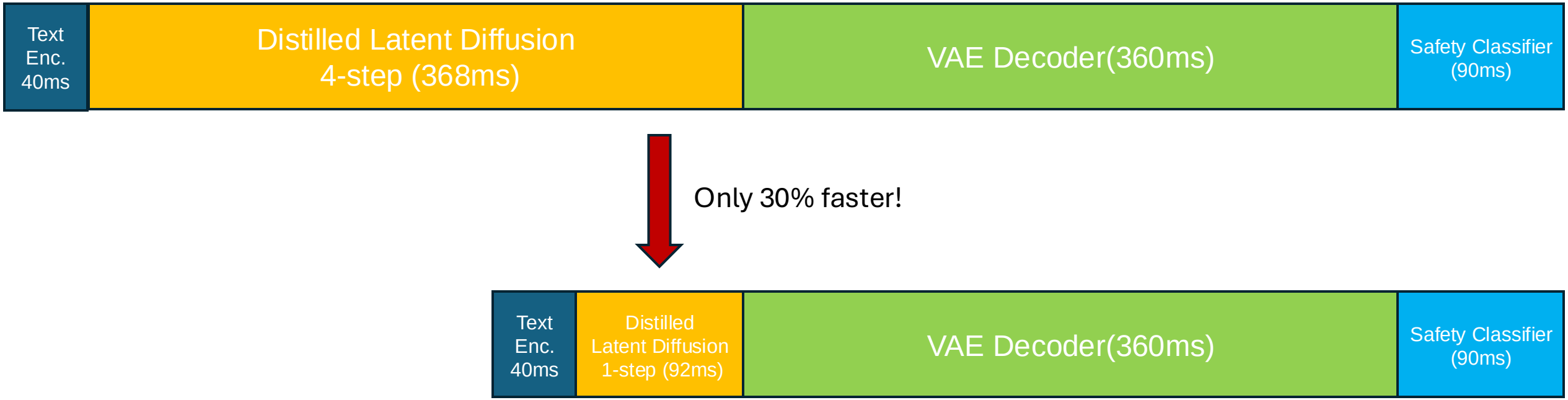
Improvement 2
Multi-step Generation

With no domain shift & quality loss, minimal additional latency

Few-steps diffusion latency



Text2Image Full Pipeline Latency Breakdown

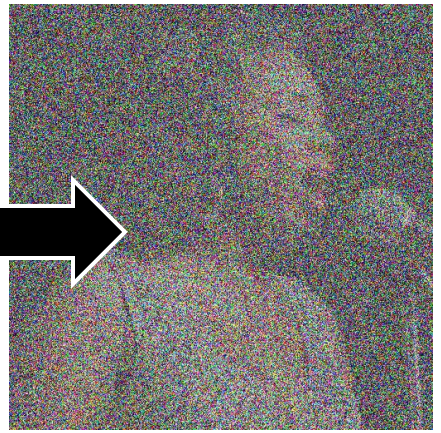



Ours 1-step (92ms)
Ours 4-step (368ms)

Domain gap Multi-Step Distillation

ground truth

training
student input (synthetic)



denoise

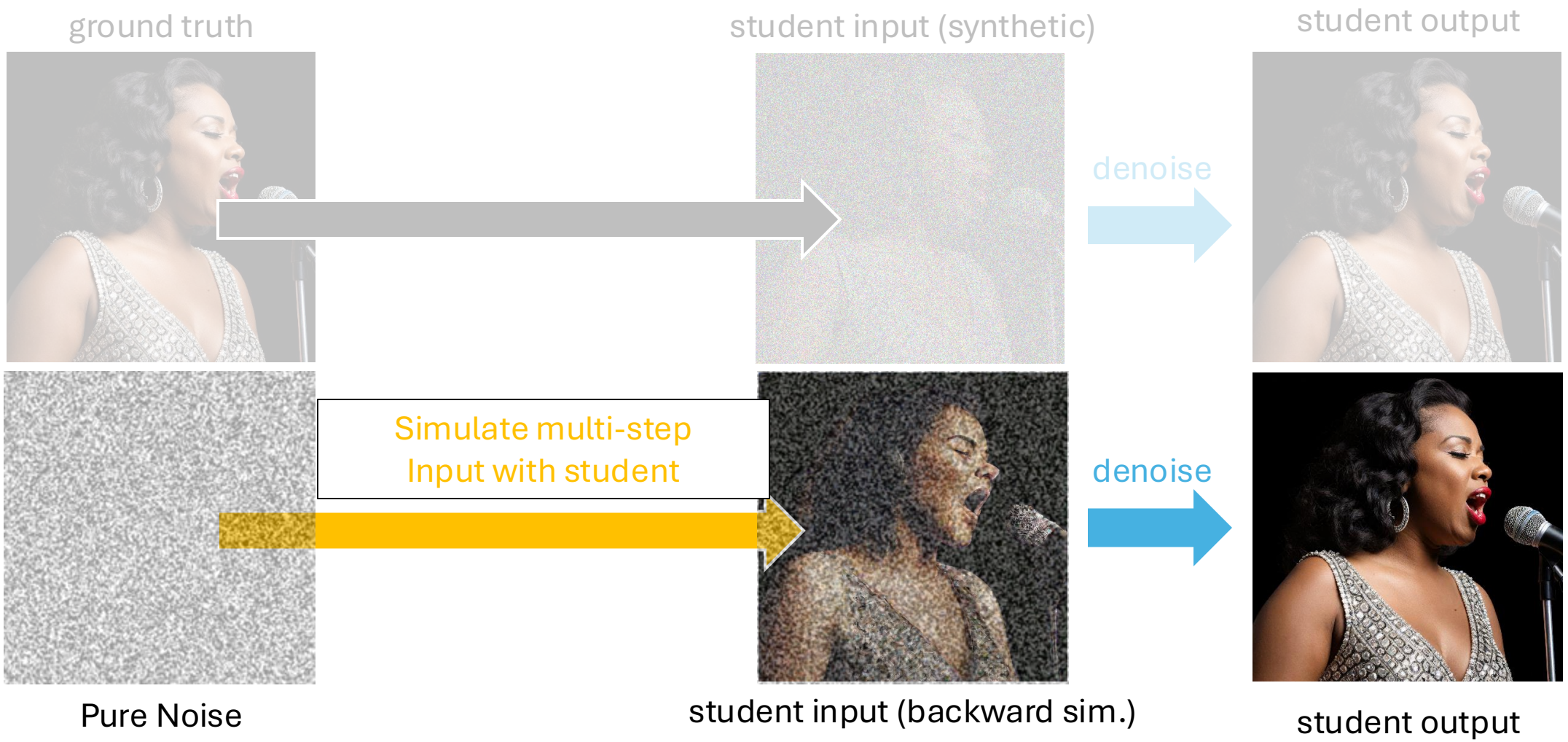
student output

inference student input

Sauer et al. 2023. Adversarial Diffusion Distillation
Sauer et al. 2024. Fast High-Resolution Image Synthesis with Latent Adversarial Diffusion Distillation
Lin et al. 2024. Sdxl-lightning: Progressive adversarial diffusion distillation.

Domain gap in Multi-Step Distillation

Fix: Backward Simulation

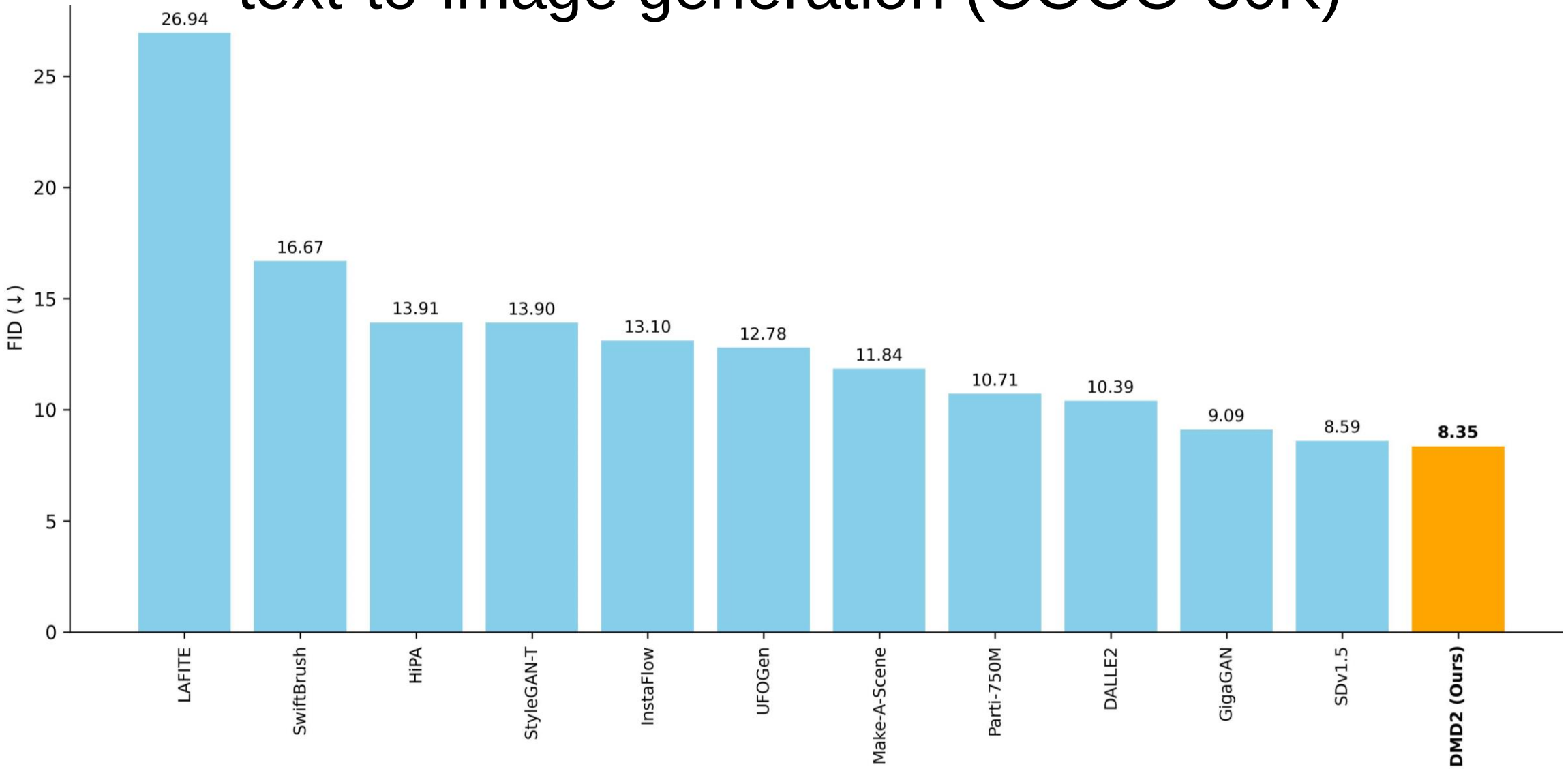



Ours w. Backward Simulation
No Backward Simulation

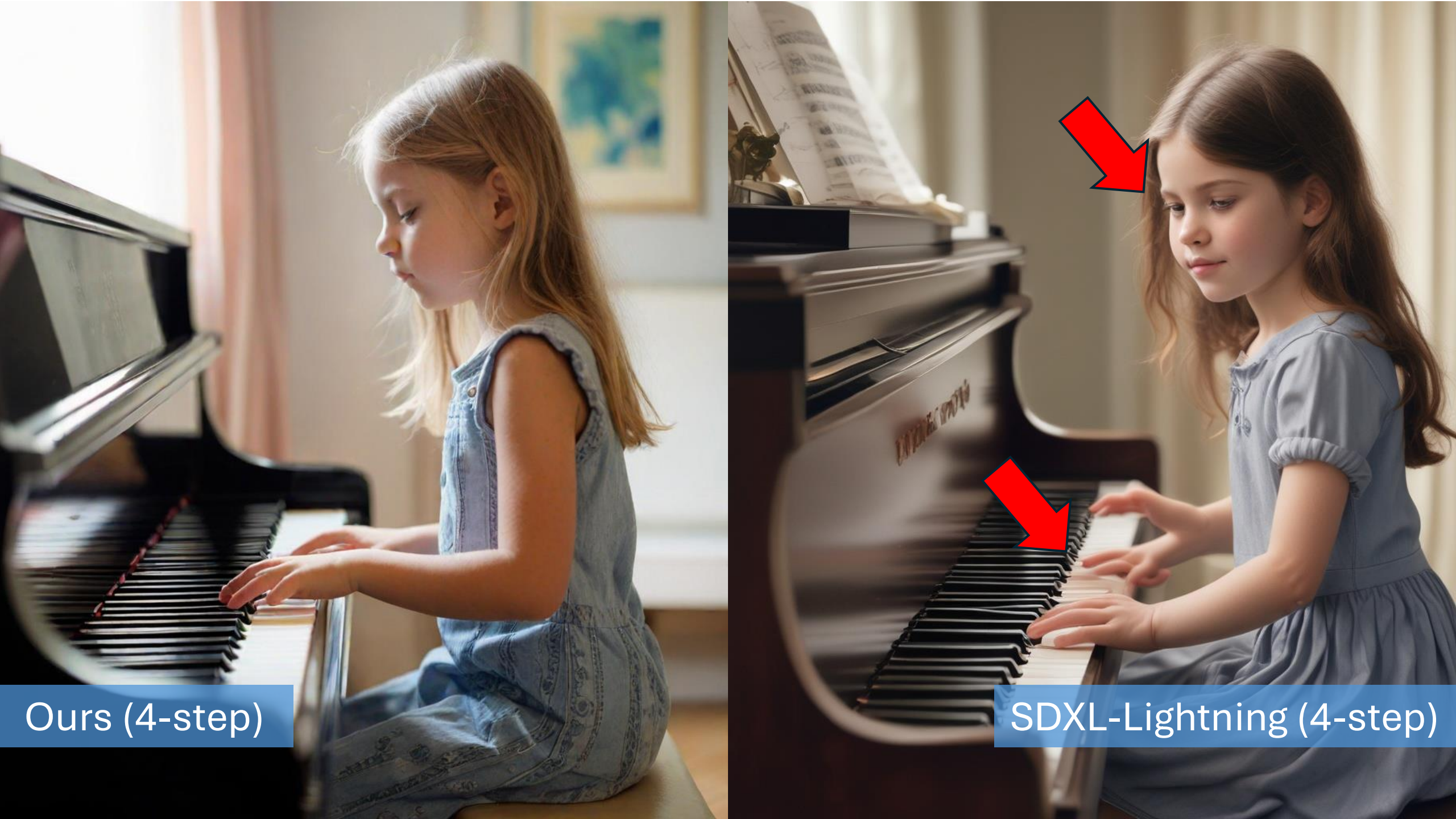
Results

“The giant magical deer god of the forest floor, [...] analagous colors, Award-winning photography”

State-of-the-art 1-step text-to-image generation (COCO-30K)




Ours (4-step)
Teacher (SDXL, 50-step)


Ours (4-step)
SDXL-Lightning (4-step)


samples from a distilled 4-step generator

Get creative with Adobe Firefly

Beautiful cozy fantasy stone cottage in a spring forest aside a cobblestone path and a babbling brook. Stone wall. Mountains in the distance. Magical tone and feel, hyper realistic.

Generate

Create with generative AI

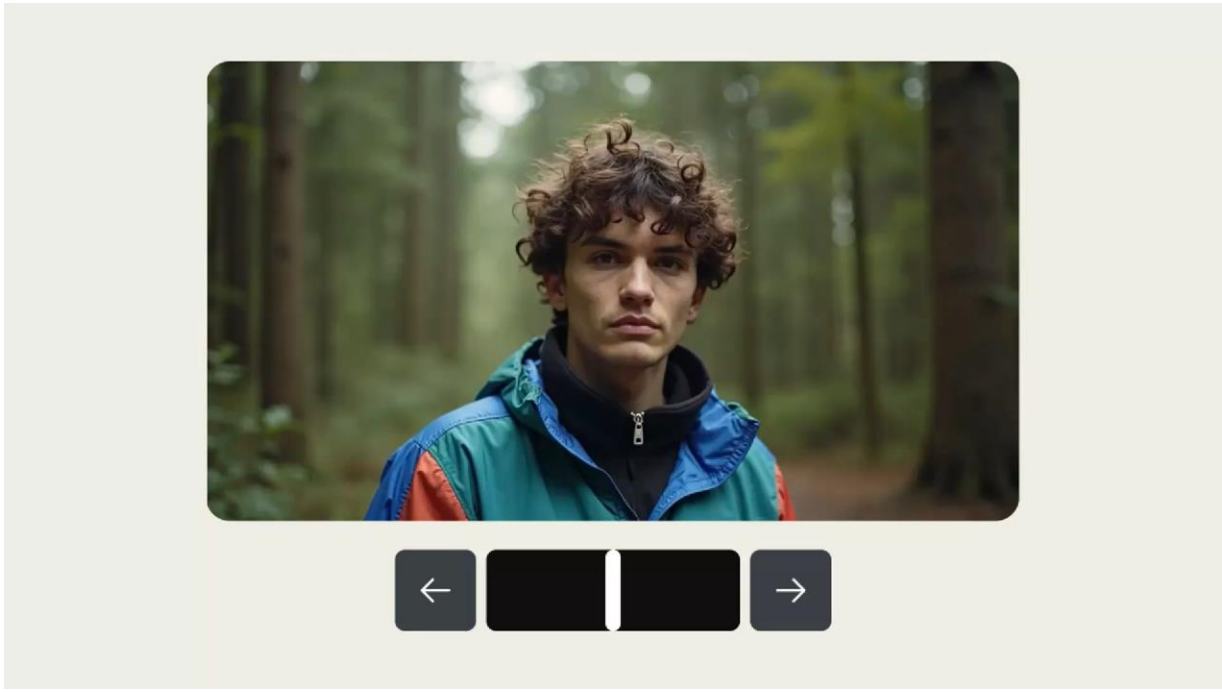
Experiment with the latest innovations from Firefly and other generative AI technology, and let us know what you think.

Video Diffusion Models

Drone view of Big Sur, generated by OpenAI's SORA

Portrait of a deer in a snowy forest, generated by Adobe's Firefly Video

Challenges in Achieving Real-Time Performance for Interactive Applications

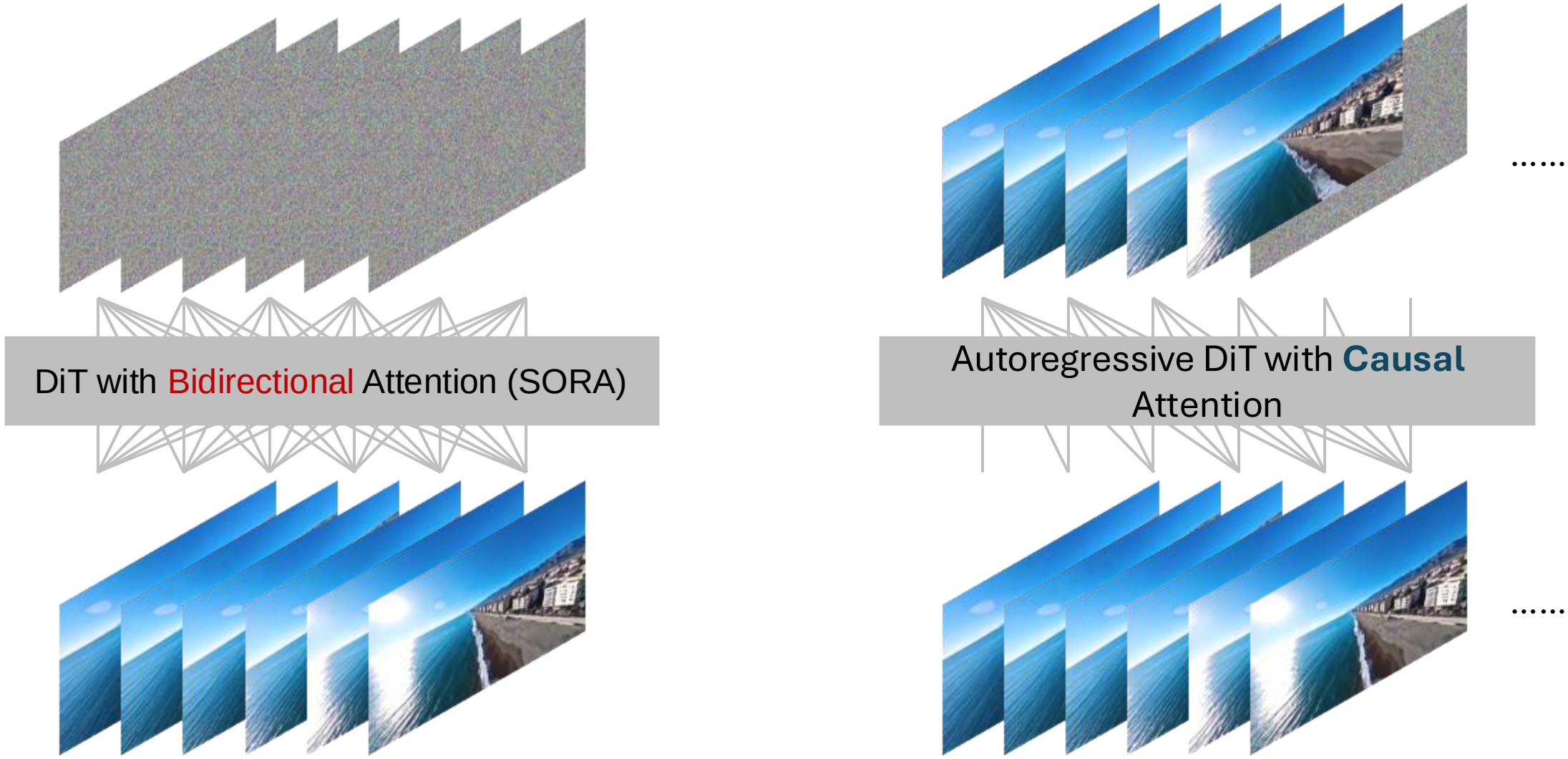


How to get the additional 10X speed up ?

Real-Time … on, video Intelligence … Demo from Runway

Combining Autoregressive Generation and Distillation for Fast Video Generation

Rethinking the Bidirectional Attention in Video Diffusion Model

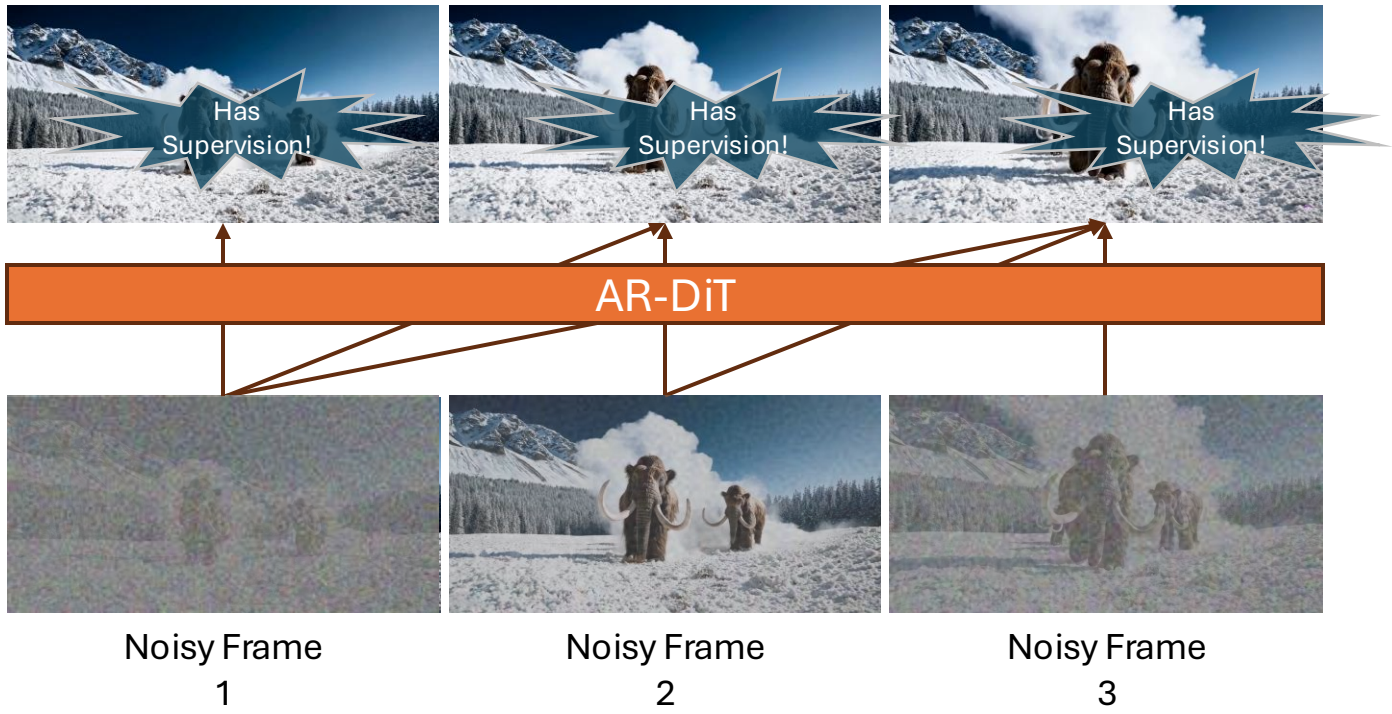


Slow, not interactive (10^3 second)

Fast, easy to interact with

Scalable Decoder-only Autoregressive Diffusion Transformer

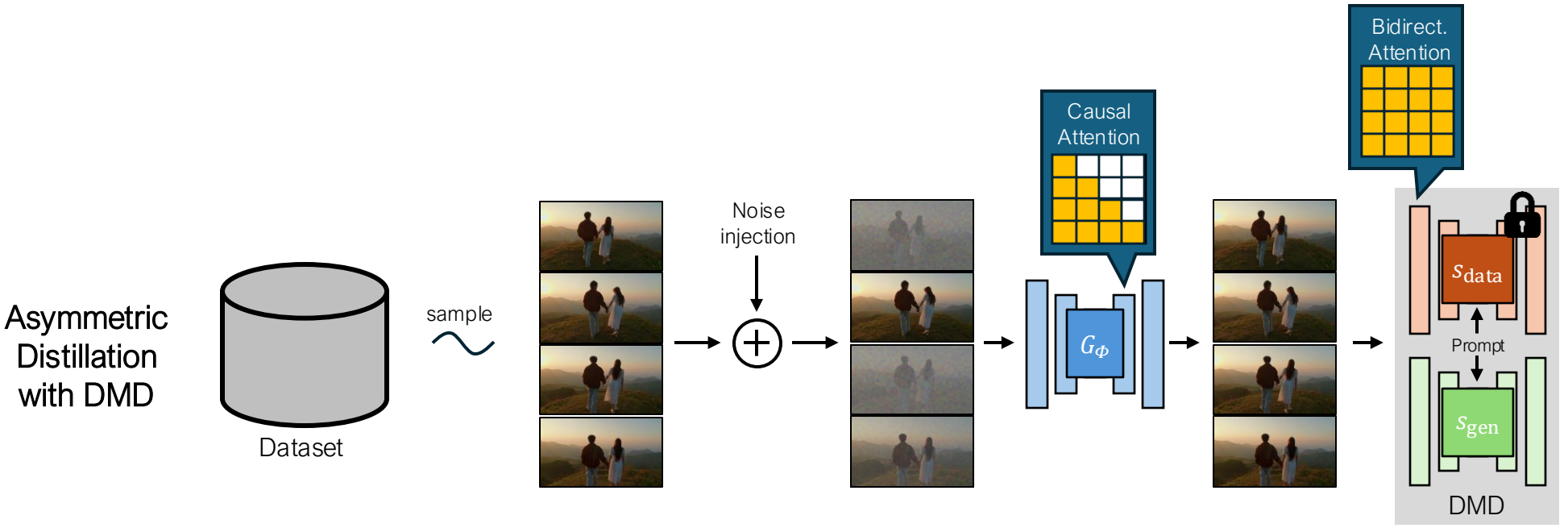
- Generalization of Diffusion-Forcing to causal DIT.
- Add noise with independent noise level to each frame.
- Given any noisy context frames, denoise all following noisy frame
- Denoising of all subsequences can be paralleled with a single forward pass using a block-wise causal mask.
- Supervision efficiency = 100%



Chen et al., “Diffusion forcing: Next-token prediction meets full-sequence diffusion”

Error Accumulation in Causal Diffusion Model

Solution: Distilling a Bidirectional Model into a Robust Causal Generator



Our Asymmetric Distillation Largely Alleviates Error Accumulation

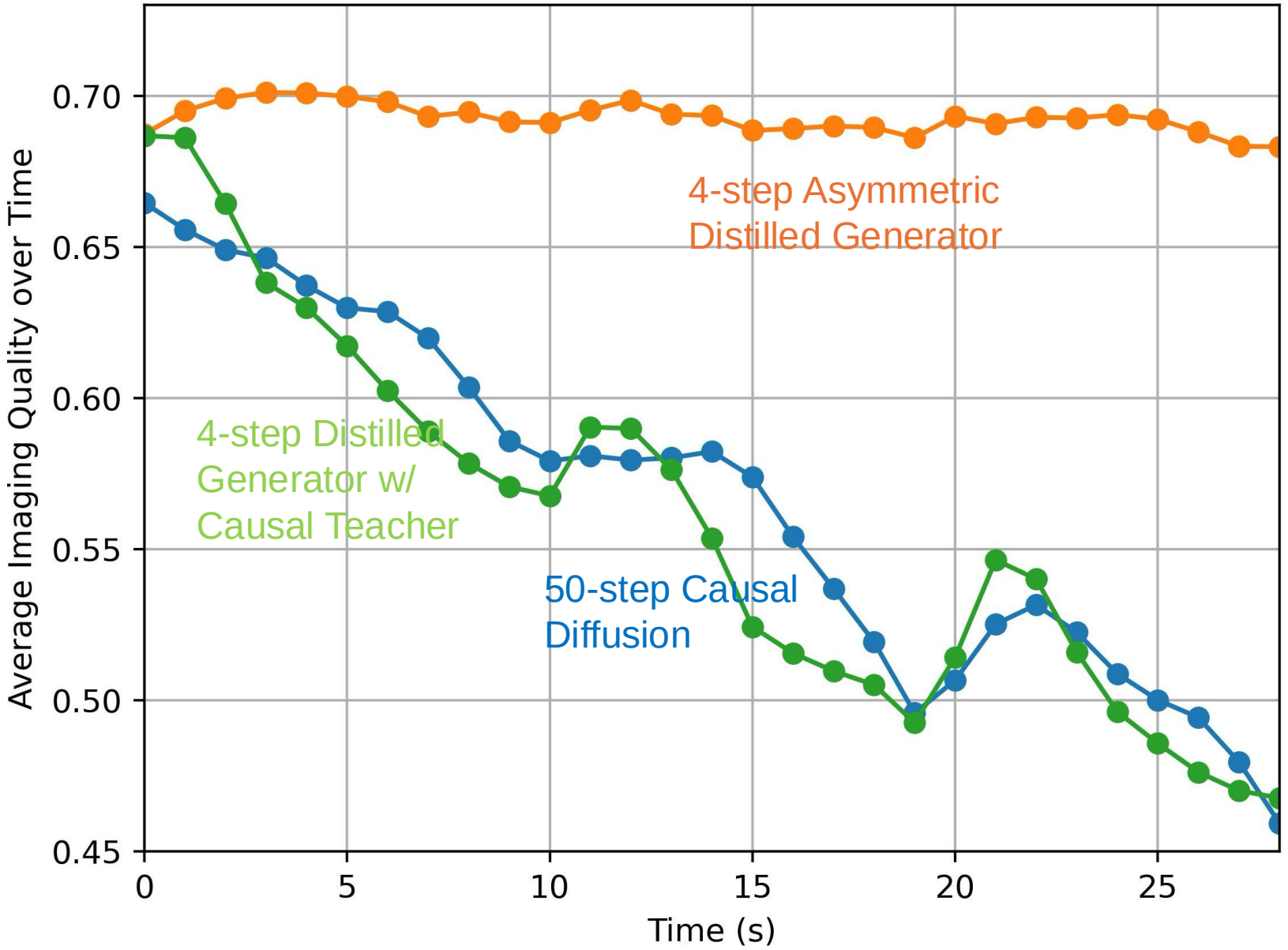
Causal Diffusion Model (50 Step)

Student Generator w/ Asymmetric Distillation (4 Step)

Distillation improves both speed and quality!

Only possible with distribution-matching style approaches!

Our Asymmetric Distillation Largely Alleviates Error Accumulation



Generators distilled from Bidirectional Teacher maintains quality over time

Intuitively, the DMD loss with a bidirectional teacher appears to be a significantly more robust objective compared to the original denoising loss.

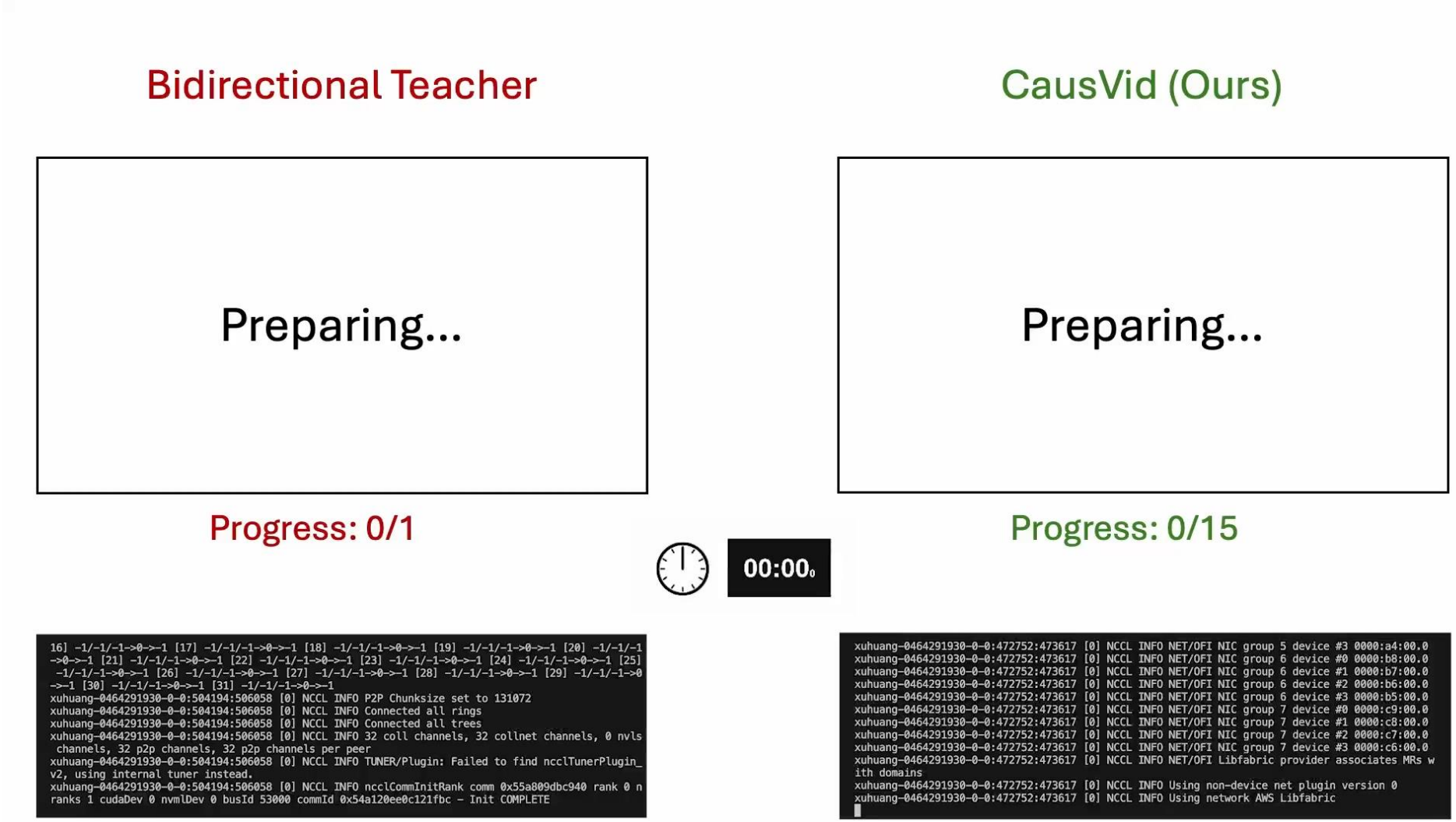
Generators distilled from causal teacher still suffers from error accumulation

Connection with LLM: Analogy of Full Sequence-Level Distribution Matching Loss and RLHF

	LLM	Autoregressive Diffusion
Pretraining	Cross Entropy Loss	Denoising Loss
	Independent for Each Token / Frame Suffer from Error Accumulation / Hallucination	
Post-Training	RLHF	DMD with bidirectional teacher
	Full-Sequence Level Loss Much more robust long-form rollout	

Results

Enabling Real-Time Video Viewing During Generation



First Place on VBench Benchmark

Model Name (clickable)	Sampled by	Evaluated by	Accessibility	Date	Total Score	Quality Score	Semantic Score
CausVid	CausVid Team	VBench Team		2024-12-07	83.88%	85.21%	78.57%
MiniMax-Video-01	VBench Team	VBench Team	API	2024-10-01	83.41%	84.85%	77.65%
HunyuanVideo (Open-Source	VBench Team	VBench Team	OpenSource	2024-12-16	83.24%	85.09%	75.82%
Gen-3 (2024-07)	VBench Team	VBench Team	API	2024-07-25	82.32%	84.11%	75.17%
Vchitect-2.0 (VEnhancer)	VBench Team	VBench Team	OpenSource	2024-09-20	82.24%	83.54%	77.06%
CogVideoX1.5-5B (5s SAT pr	VBench Team	VBench Team	OpenSource	2024-11-15	82.17%	82.78%	79.76%
Jimeng	VBench Team	VBench Team	API	2024-11-15	81.97%	83.29%	76.69%
Vidu	VBench Team	VBench Team	API	2024-11-15	81.89%	83.85%	74.04%
Kling (2024-07 high-perfor	VBench Team	VBench Team	API	2024-08-01	81.85%	83.39%	75.68%
CogVideoX-5B (SAT prompt-o	VBench Team	VBench Team	OpenSource	2024-09-02	81.61%	82.75%	77.04%
Vchitect-2.0-2B	VBench Team	VBench Team	OpenSource	2024-09-16	81.57%	82.51%	77.79%
CogVideoX-2B (SAT prompt-o	VBench Team	VBench Team	OpenSource	2024-08-19	80.91%	82.18%	75.83%
Pika-1.0 (2024-06)	VBench Team	VBench Team	API	2024-07-29	80.69%	82.92%	71.77%

Text to Video Generation

“Macro shot of a man wearing an antique diving helmet with dark glass and a jetpack walking on the veins of a leaf. Realistic style”

Text to Video Generation

“A Samoyed and a Golden Retriever dog are playfully romping through a futuristic neon city at night. The neon lights emitted from the nearby buildings glistens off of their fur.”

Text to Video Generation

“a spooky haunted mansion, with friendly jack o lanterns and ghost characters welcoming trick or treaters to the entrance, tilt shift photography.”

Long Video Generation via Sliding Window Inference

“A bear made of strawberrys is walking in the forest, its eyes looking around as if it is seeing the world for the first time.”

Zero-Shot Image to Video Generation

“Rocket blasting off from a laptop screen on an organized office table. The rocket leaves the screen and blast into space.”

Limitation: Reduced Diversity

Macro cinematography, slow motion shot: A sculptor's hands shape wet clay on a wheel, and as the wheel spins. Camera captures the tactile quality of the clay and the fluid motion of the sculptor’s hands.

50-step Bidirectional Diffusion Model

4-step Causal Generator

Discussion

Trajectory Preserving

+ Better diversity

- Blurry outputs for few-step generation

- Over constrained with limited performance upper bound

- Teacher/student often requires same architecture

Mix of Both

Hyper-SD, DMD1, SDXL-Lightning

Explorations on standalone methods:

Moment Matching Distillation (DM)

Continuous-Time Consistency Models (TP)

Distribution Matching

- Mode collapse

+ Higher quality for few-step generation

- More unstable training

+ More flexible architecture configuration